CONTROL DATA CORPORATION

NETWORK PRODUCTS

**NETWORK ACCESS METHOD
VERSION 1
NETWORK DEFINITION LANGUAGE
REFERENCE MANUAL**

**CDC® OPERATING SYSTEM:
NOS 2**

REVISION RECORD

Revision	Description
A (12/01/76)	Original release. Programming System Report (PSR) level 439.
B (04/01/77)	Revised at PSR level 446 for technical corrections.
C (04/28/78)	Completely revised for NAM Version 1.1 release at PSR level 472 to include support of remote and foreign NPUs, asynchronous and HASP TIPs, virtual terminals, IAF, and TVF.
D (08/15/78)	Revised at PSR level 477 for technical corrections.
E (12/18/78)	Revised at PSR level 485 to include autorecognition up to 1200 baud and for technical corrections.
F (08/10/79)	Revised to reflect release of NAM Version 1.2. Included are descriptions of special editing support, extended APL support, and various technical corrections.
G (05/23/80)	Revised at PSR level 517 to reflect support of the 714-30 terminal class, and of 714-10/20 and 714-30 line printers. Various technical corrections are also included.
H (10/31/80)	Revised at PSR level 528 to document support of the X.25 protocol.
J (05/29/81)	Revised to reflect release of NAM Version 1.3 at PSR level 541 to document support of the PRU interface and the BSC protocol (2780/3780 terminal support). Also includes various technical corrections.
K (12/22/82)	Revised to reflect release of NAM Version 1.5 (Version 1.4 does not exist) at PSR level 580, which supports the Network Access Method and Communication Control Program Version 3.5 under NOS Version 2; this manual no longer applies to NOS Version 1 systems. This is a complete reprint.
L (09/30/83)	Revised to reflect release of NAM Version 1.6 at PSR level 599, which supports the Network Access Method and Communications Control Program Version 3.6 under NOS Version 2. This is a complete reprint.
M (09/19/84)	Revised to reflect release of NAM Version 1.7 at PSR level 617, which supports the Network Access Method and Communications Control Program Version 3.7 under NOS Version 2.3. Support of the CYBER 170 800 Series models and the CYBER 180 Computer Systems is documented.
N (09/30/85)	Revised to reflect release of NAM Version 1.8 at PSR level 642, which supports the Network Access Method and Communications Control Program Version 3.8 under NOS Version 2.4.2. Miscellaneous technical changes are included. This is a complete reprint.
P (12/16/85)	Revised to reflect release of NAM Version 1.8 at PSR level 647, which supports the Network Access Method and Communications Control Program Version 3.8 under NOS 2.4.3 and the release of CDCNET Version 1.0. Miscellaneous technical changes are included.
R (07/31/86)	Revised to reflect release of NAM Version 1.8 at PSR level 664, which supports the Network Access Method and Communications Control Program Version 3.8 under NOS 2.5.1 and the release of CDCNET 1.1. Miscellaneous technical changes are included.
S (04/23/87)	Revised to reflect release of NAM Version 1.8 at PSR level 678, which supports the Network Access Method and Communications Control Program Version 3.8 under NOS 2.5.2 and the release of CDCNET 1.2. Three new parameters have been added to the OUTCALL statement and changes have been made to default values for other parameters to simplify connections to NOS/VE. Miscellaneous technical changes are included.

REVISION LETTERS I, O, Q, AND X ARE NOT USED

60480000 S

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

PREFACE

This manual describes the Network Definition Language (NDL) for the CONTROL DATA® Network Access Method (NAM), Version 1.8. It assumes that the reader is a network site administrator familiar with the Network Operating System (NOS) and other software in the networks product set.

The Network Access Method Version 1.8 operates under control of the NOS 2 operating system for the CDC® CYBER 180 Series; CYBER 170 Series; CDC CYBER 70 Models 71, 72, 73, and 74; and 6000 Series Computer Systems.

The Network Definition Language processor is a compiler used by a network administrator to create and maintain the files that define the physical and logical structure of the network for other network software, which in turn establishes, initializes, and operates the network.

RELATED PUBLICATIONS

Related material is contained in the Control Data Corporation publications listed below. The publications are listed within groupings that indicate relative importance to readers of this manual.

The NOS System Information Manual is an online manual that includes brief descriptions of all NOS and NOS product manuals. To access this manual, log in to NOS and enter the command EXPLAIN.

The following publications are of primary interest:

Publication	Publication Number
CDCNET Conceptual Overview	60461540
CDCNET Configuration and Site Administration Manual	60461550
CDCNET Systems Programmer's Reference Manual Volume 1, Base System Software	60462410
CDCNET Systems Programmer's Reference Manual Volume 2, Network Management Entities and Layer Interfaces	60462420
CDCNET Systems Programmer's Reference Manual Volume 3, Network Products	60462430
CDCNET Systems Programmer's Reference Manual Volume 4, Terminal Interface Programs	60462440
CDCNET Terminal Interface Usage Manual	60461530
CYBER Cross System Version 1 Build Utilities Reference Manual	60471200
Network Products Network Access Method Version 1 Host Application Programming Reference Manual	60499500
Network Access Method Version 1/ Communications Control Program Version 3 Terminal Interfaces Reference Manual	60480600
NOS Version 2 Reference Set, Volume 3, System Commands	60459680
NOS Version 2 Analysis Handbook	60459300

The following publication is of secondary interest:

Publication	Publication Number
NOS Version 2 Installation Handbook	60459320

Sites within the United States can order CDC manuals from Control Data Corporation, Literature and Distribution Services, 308 North Dale Street, St. Paul, Minnesota 55103.

Other sites can order CDC manuals by contacting the local sales office.

> This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

Also, if you have access to SOLVER, the CDC online facility for reporting problems, you can use it to submit comments about this manual. When it prompts you for a product identifier for your report, please specify NA5.

CONTENTS

APPENDIXES

INDEX

FIGURES

TABLES

NOTATIONS

Throughout this manual, the following conventions are used to present statement formats and diagnostic messages:

[ ] Square brackets indicate that the enclosed parameters or values are optional. When two or more items are stacked vertically within brackets, one of them can be used or all may be omitted. An entire parameter or part of a parameter can be optional. For example, the brackets in [CP=cp] indicate that the whole parameter is optional and can be omitted, but the brackets in AUTO [=yn1] show that you can specify the parameter as either AUTO=yn1 or AUTO only.

___ Underlines indicate defaults. If the omission of any entity causes a single default to be used, the default value is underlined.

{ } Braces enclosing vertically stacked items indicate that one of the enclosed items is required and that only one can be chosen. When horizontally arranged parameters are enclosed in braces, one or more of the parameters can be used, but at least one is mandatory.

... Ellipses indicate that omitted entities repeat the form and function of the last entity given. An ellipsis immediately following a parameter indicates it can be repeated at your option.

UPPERCASE Uppercase letters indicate words, acronyms, or mnemonics either required by the NDL processor or produced as output. All words printed entirely in uppercase letters have a preassigned meaning to the NDL processor. These words include statement identifiers, keywords, and reserved word values.

lowercase Lowercase words identify variables for which values are supplied by you or by the system as output. These words generally indicate the nature of the information they represent (numerical value, file or job name, and so forth).

Δ The delta symbol represents a blank used as a separator. Anywhere a delta symbol is shown, a comma can be used. Multiple blanks or a blank following a comma is ignored. Multiple commas are illegal, except when they separate unused optional parameters in the login procedure.

[cr] The boxed cr symbol represents the terminal key that ends a message; usually, this is the same key that causes a carriage return operation.

LF The LF symbol represents a one-line vertical repositioning of the cursor or output mechanism. LF also designates a character or character code associated with such a line feed operation.

Ⓒ A circle around a character represents a character key that is pressed in conjunction with a control key (CTL, CNTRL, CONTRL, CONTROL, or equivalent).

Unless otherwise specified, all references to numbers are to decimal values; all references to characters are to 6-bit display coded characters.

As a site administrator, you must code and run a job that uses the CDC Network Definition Language to describe the hardware and software elements that comprise the computer and communication network for your site. A Network Definition Language (NDL) job creates two types of network definition files: network configuration files and local configuration files.

The network configuration file contains information about the physical and logical configuration of the network. The local configuration file contains information about the network application programs that run in a host computer and provide services to the network, defines login options for the devices that access a host computer, and defines connections that applications can have to other hosts.

Only the local configuration file applies to Control Data Distributed Communications Networks (CDCNET). For information on configuring CDCNET networks, refer to the CDCNET Configuration and Site Administration Manual.

NETWORK DEFINITION LANGUAGE PROCESSOR

Your job processes source statements through the Network Definition Language processor. The NDL processor creates the network definition files, as shown in figure 1-1.
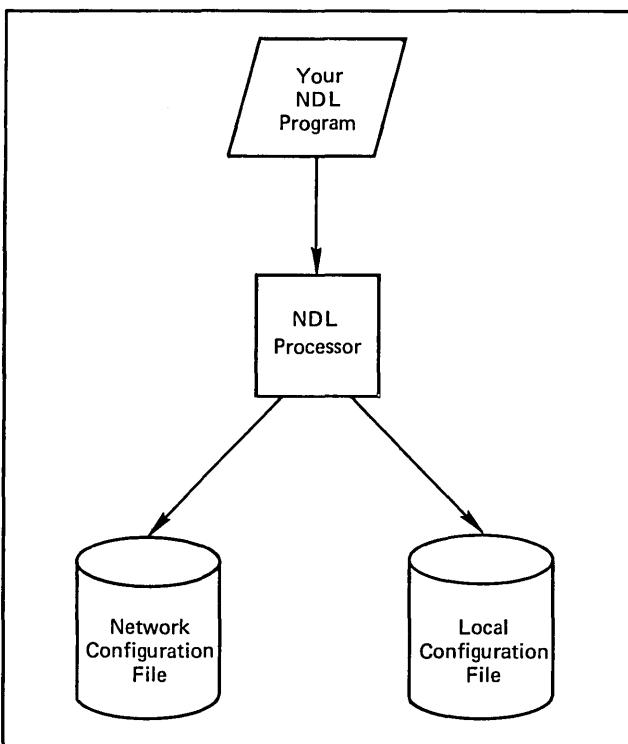
Figure 1-1. Creation of Network Definition Files

In addition to the two types of network definition files, the NDL processor produces a job listing file. You can make the processor provide the following information in the listing file:

An annotated copy of the NDL program source statements input to the processor

Descriptions of errors noted on a source statement listing

Descriptions of the network configuration file and local configuration file contents

A list of all legal DEFINE statements (statements used like macros to equate a character string with an identifier for later insertion in one of the definitions)

An annotated copy of the NDL program source statements input to the processor, with all DEFINE statement character strings inserted in place of identifiers

The listing file contents are described further in section 10, Job Structure.

The NDL processor executes as a batch preprocessing compiler; it does not interact with the network during network operation.

BASIC NETWORK CONCEPTS

The following concepts will help you understand the relationships among the hardware and software elements in your network. A more detailed description of the elements of a CDC network is presented at the end of this section.

CDC network software supports the seven layers of protocols defined in the International Standards Organization Model for Open System Interconnection (ISO/OSI), as described in document TC97/SC16/N227 or N309. References in this manual to layer numbers are references to the layer definitions in that document. The ISO/OSI layers are:

Layer 1 - Protocols defining the physical and electrical characteristics of connections between two components of the network

Layer 2 - Protocols for physical linkages that connect elements within the network

Layer 3 - Protocols for logical linkages that connect elements through the network

Layer 4 - Protocols for physical data transport within the network

Layer 5 - Protocols for managing data transport through the network

Layer 6 - Protocols for data formats between elements connected to the network

Layer 7 - Protocols for element supervision within the network

You do not need to know the protocols in these seven layers. However, some of the things you must include in your network definition are determined by the needs each protocol imposes on the network software.

You can think of a CDC network as a hierarchy of hardware and software elements with three functional levels. Each functional level implements one or more of the ISO/OSI protocol layers.

The separated portions of these three levels are shown in figure 1-2. The outermost level is the service network. The service network comprises the site's local service elements. The local service elements include the network application programs in a host computer and the people at terminals using those programs.

```
┌─────────────────────────────────────────────┐
│      ╱───────────────────────────────╲      │
│     ╱    Local Service Elements        ╲     │
│    ╱     (Application Programs)          ╲    │
│   ╱───────────────────────────────────────╲  │
│   ╲     Computer Elements                 ╱  │
│    ╲    (Host Computer and               ╱   │
│     ╲   Supervisory Programs)           ╱    │
│      ╲─────────────────────────────────╱     │
│      ╱    Communication Elements        ╲    │
│     ╱     (Connected Concentrator         ╲   │
│    ╱      Hardware and Software)           ╲  │
│    ╲──────────────────────────────────────╱  │
│     ╲    Computer Elements              ╱    │
│      ╲   (Terminal Hardware)           ╱     │
│       ╲───────────────────────────────╱      │
│      ╱    Local Service Elements       ╲     │
│     ╱     (Terminal Users)               ╲    │
│    ╱─────────────────────────────────────╲   │
└─────────────────────────────────────────────┘
```
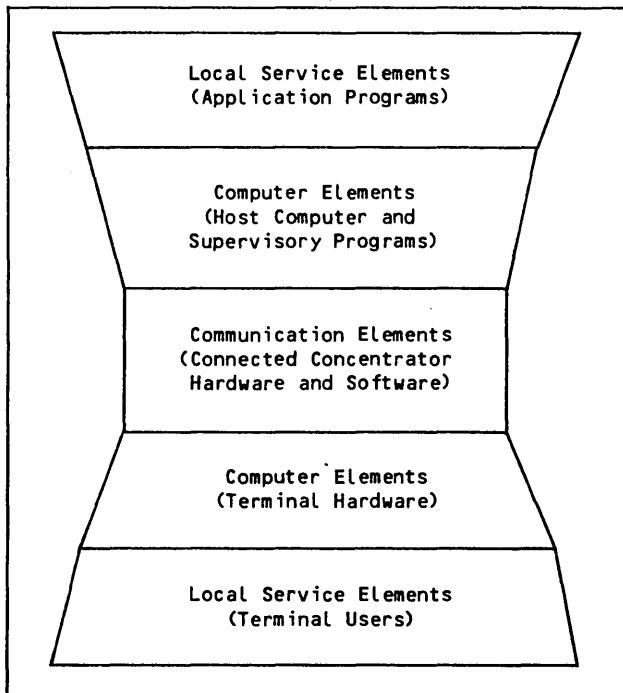
Figure 1-2. Element Levels Within a Network

The middle level is the computer network. The computer network comprises the site-administered local and remote computer elements. The computer elements include the terminals, a host computer, and network software supervisory programs in a host computer.

The innermost level is the communications network. The communications network comprises the local and remote communications elements. The communications elements include the communications concentrator hardware (called network processing units, or NPUs), the concentrator software, and the linkages (ISO/OSI layers 2 and 3) among these elements needed to connect the isolated ends of the other levels.

To ensure proper functioning of a CDC network, you must configure the functional levels correctly.

Using NDL source statements, you provide the information necessary to identify the hardware and software elements in each level of the network to the other levels of the network. You must also establish logical relationships among the hardware and software elements.

SUPERVISORY PROGRAMS

Each functional level of the network hierarchy has corresponding supervisory software. CDC provides three supervisory programs:

The Network Validation Facility

The Communications Supervisor

The Network Supervisor

These supervisory programs run in a host computer and use the network definition files to initialize, monitor, and control network operations (ISO/OSI layer 7). Figure 1-3 shows the administrative files used by these supervisory programs.

The Network Validation Facility (NVF) uses the local configuration file in conjunction with the NOS system validation file (VALIDUz) to determine which host resources terminal users are allowed to access and which applications can run in the network. NVF runs in each host in the network.

The Communications Supervisor (CS) uses information from the network configuration file to monitor and control NPU and device operations (ISO/OSI layer 5). CS oversees the continuing operations of the communication elements and the computer elements outside of the site host. CS can run in one or more hosts in the network.

The Network Supervisor (NS) uses information from the network configuration file in conjunction with the NPU load file to load network processing units. The information NS loads determines the initial operation of the communication elements. NS can run in one or more hosts in the network.

NETWORK ACCESS METHOD INTERFACES

All data messages pass through the Network Access Method (NAM) to ensure their integrity as they are routed among terminals, the host computer, and applications. NAM consists primarily of three interface programs (ISO/OSI layers 4 and 5) in the host computer:

The Peripheral Interface Program

The Network Interface Program

The Application Interface Program

The Peripheral Interface Program runs in a host computer peripheral processor. The Network Interface Program runs in the host computer's central processor and uses a system control point; it is thus able to communicate with other programs using other control points. A copy of the Application Interface Program runs in the field length of every network application program, including the supervisory programs. Figure 1-4 shows the relationships among this software.
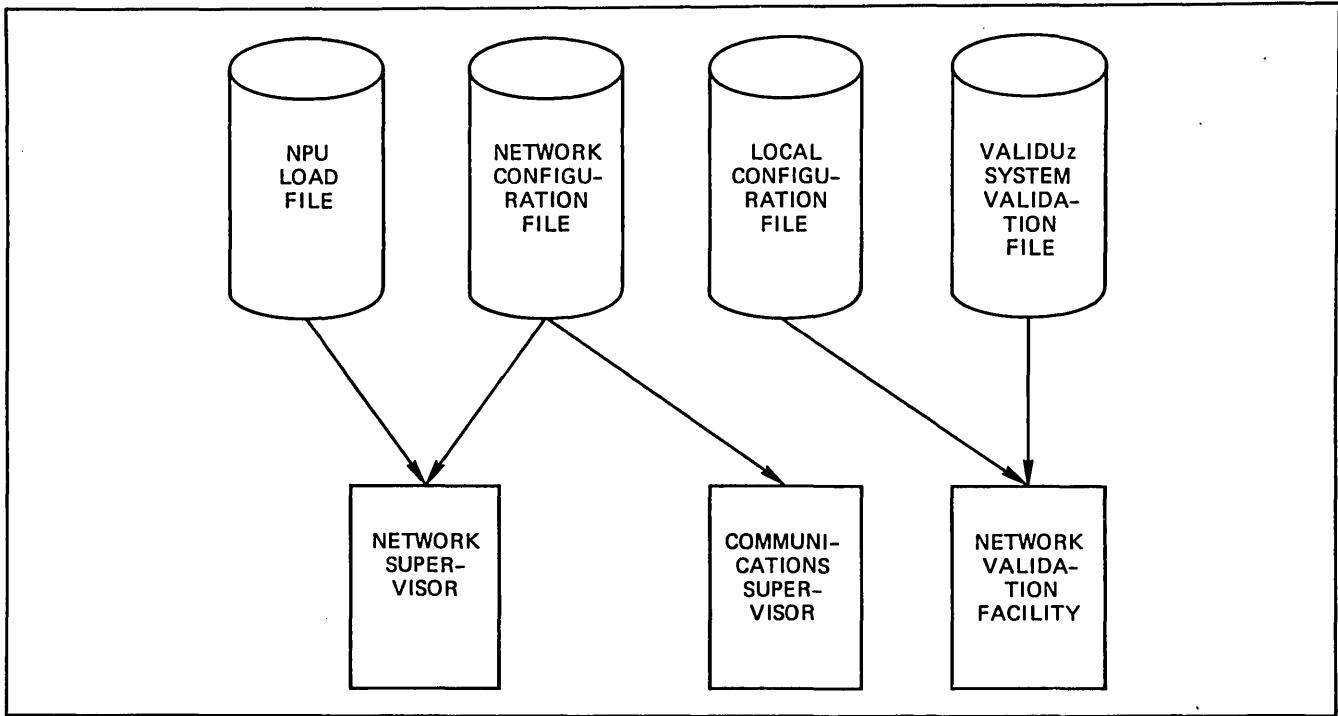
Figure 1-3. Supervisory Programs and File Use
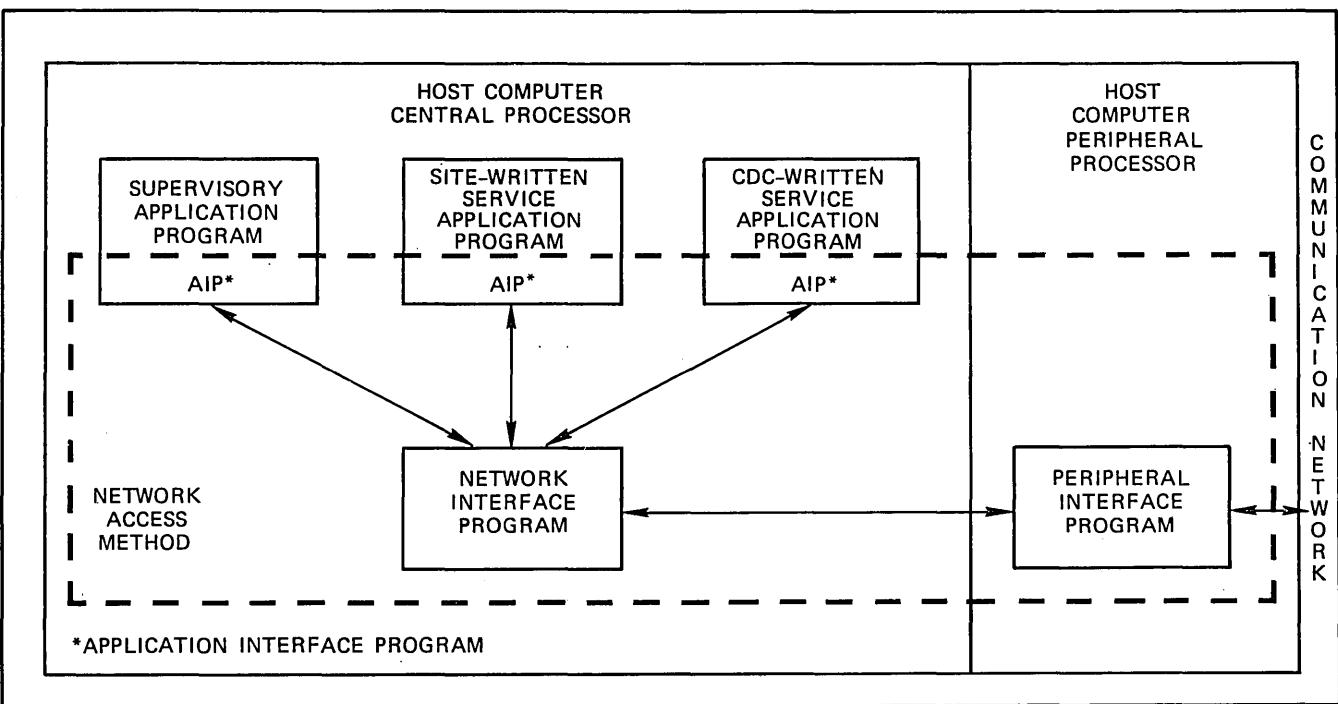
Figure 1-4. The Network Access Method

The supervisory application program in figure 1-4 could be one of the three CDC-written programs shown in figure 1-3, or it could be a site-written program tailored to your needs. You can provide your own network service application programs, or you can use CDC-written service programs such as the Interactive Facility or the Remote Batch Facility. The CDC-written network service programs are described later in this section.

SIMPLE NETWORK

Figure 1-5 shows a possible network, Simnet. In this simple network, a single host computer, a CDC CYBER 170 Model 176, is connected to a network processing unit; the NPU is a small, independent processor. This connection consists of a CYBER 170 host computer data channel cabled to a hardware module (called a coupler) in the NPU. An NPU connected to the host in this manner is known as a front-end NPU. The front-end NPU in Simnet is NPU1, a CDC 255x Communications Processor.
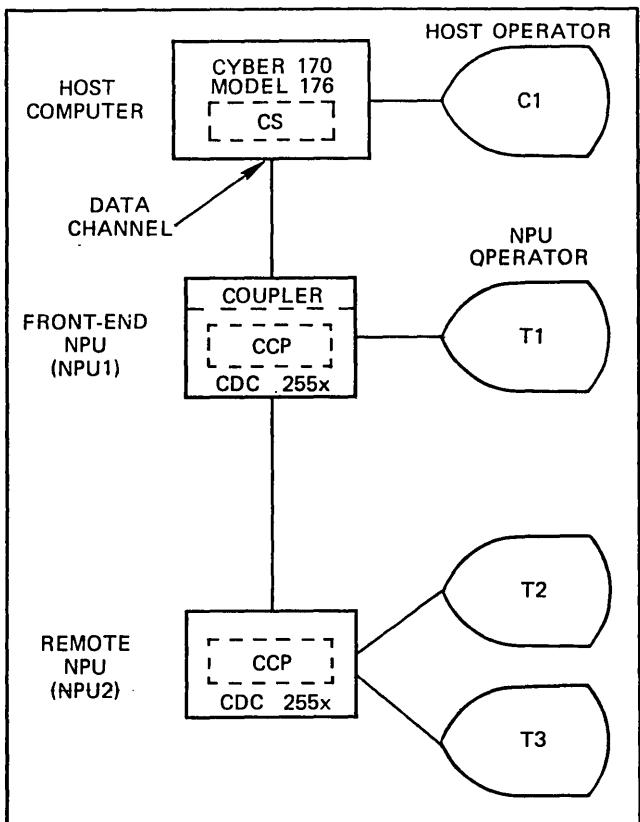
Figure 1-5. Simnet - A Simple Configuration

The front-end NPU can support terminals and other NPUs, known as remote NPUs. Remote NPUs can also support terminals. The only remote NPU in this network is NPU2, a CDC 255x Communications Processor.

The front-end network processing unit NPU1 supports terminal T1, and the remote processing unit NPU2 supports terminals T2 and T3. CDC provides support for many different types of terminals and terminal devices, as described in sections 4 through 7.

In the figure, C1 is the host computer's system console. The system console is not considered a network terminal and therefore is not defined in an NDL program. The person at the system console, however, is an administrative operator for the network.

Administrative Operators

A CDC network supports three types of administrative operators:

The host operator (HOP)

The NPU operator (NOP)

The diagnostic operator (DOP)

The host operator is located at the system console for the site's host computer. The HOP can monitor and control application status, monitor NPU dump and load status, plus perform all of the functions of an NPU operator. The HOP always has control over all of the network.

The NPU operator is located at a terminal or host computer console. The NOP can obtain and change the status of network elements, communicate with terminal users, and run diagnostic tests. Each NPU can have one controlling NOP.

The diagnostic operator is located at a terminal serviced by the network. The DOP can monitor the status of network elements and can run diagnostic tests on an NPU. Any terminal user with permission to log in to the Communications Supervisor can become a DOP. There can be several DOPs active at the same time.

In Simnet, the host operator is located at the system console (C1), and an NPU operator is located at terminal T1. Both administrative operators communicate with the Communications Supervisor in the host computer.

Variants of NPU Software

Each NPU, because it is an independent processor, has its own software. The software an NPU requires to access the network is called its variant. You must describe each NPU's software to the supervisory programs. You do so by identifying which program initiation control block (PICB) within the NPU load file properly describes the software in each NPU.

CDC 255x NPUs require loading, configuring, and operational control from the supervisory programs in the host. NPUs that can load themselves but require host configuration and supervision to operate or that are self-loading and self-configuring can also belong to the network.

The software variant used in a particular NPU depends on various aspects of the network to be configured, including:

The type of NPU and host hardware in your network (CDC or non-CDC equipment, front-end or remote NPU, and so forth)

The network topology (connections between front-end and remote NPUs)

The types of terminals and devices the NPU must support

The software in a CDC 255x series Communications Processor network processing unit is collectively called the Communication Control Program (CCP). CCP can contain three types of interface programs:

A Host Interface Program (HIP)

A Link Interface Program (LIP)

One or more Terminal Interface Programs (TIPs)

A HIP is needed only if the CCP variant links a front-end NPU to a host computer (ISO/OSI layer 2). A LIP is needed only if the CCP variant links a remote NPU and a front-end NPU (ISO/OSI layer 2). A TIP is needed only if the CCP variant links terminals to the network or an X.25 application-to-application connection (ISO/OSI layers 2, 4, and 6).

Figure 1-6 presents another view of Simnet, illustrating the relationships among these interface programs. Note that terminals T2 and T3 each connect to a different Terminal Interface Program. T2 is a CDC 752, which is an asynchronous terminal; T3 is a CDC 200 User Terminal, which is a synchronous terminal. Each has different support requirements; therefore, a different interface program is required for each.

The network software currently supports TIPs for five general terminal protocols. Your site can provide its own Terminal Interface Program software to support up to three additional protocols; your site can also modify CDC-written Terminal Interface Programs to support local variants of the five general protocols.

PACKET-SWITCHING NETWORK

CDC network software provides support for asynchronous terminals connected to public packet-switching networks (PSNs). This support complies with recommendation X.25 of the International Telegraph and Telephone Consultative Committee (CCITT) for standardizing the interface between data terminal equipment and packet-switching networks. (Packet-switching networks are sometimes called public data networks, or PDNs.)

Figure 1-7 shows the packet assembly/disassembly (PAD) service, which is one part of the interface between the network and the data terminal equipment. This part of the interface is provided by the packet-switching network.

Figure 1-6. CCP Interface Programs
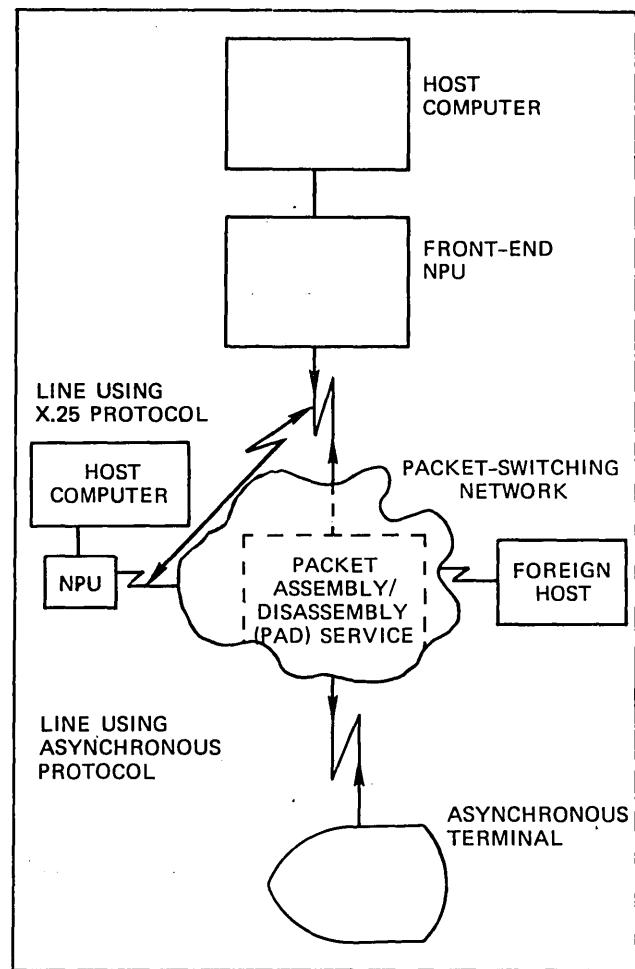
Figure 1-7. Packet-switching Network Interface

Since the X.25 protocol is not symmetric, the roles played by each end of an X.25 link must be identified for each X.25 line. The ends of a X.25 link are the data terminal equipment (DTE) and the data circuit equipment (DCE).

A calling terminal is linked with the Communication Control Program in the NPU via a logical path called a virtual circuit. The site must subscribe to the packet-switching network for the desired number of virtual circuits as well as other parameters (described in section 5) affecting the operation of the NPU-PSN interface. CDC support of this type of configuration is described in detail in section 5.

MULTIPLE-HOST NETWORK

Network software supports networks with multiple host computers. A simple multiple-host configuration is illustrated by Dualnet in figure 1-8. A second host has been added to the network, connecting to a coupler in NPU2. Both NPU1 and NPU2 are channel-connected as front-end NPUs to a local host and are connected to a remote host via a trunk communication line.

A more complicated multiple-host network concept is illustrated by Multinet in figure 1-9. Note that each side of the figure resembles Simnet in figure 1-5. In effect, two entire networks have been merged. This has been accomplished by connecting each front-end processor to both hosts. Figure 1-7 shows a multiple-host network where the hosts are interconnected by an X.25 packet-switching network.

Multiple-host networks offer three features that are not available in single-networks. These are:

The ability of a terminal user to connect to one of many hosts.

The ability of an application in one host to communicate with an application in one or more other hosts.

The ability of the NPUs in the network to be loaded, configured, and supervised by one of several hosts in the network.

The installation can either preassign terminal users to hosts, or can allow them to select a host when they connect to the network. In both cases, terminal users can either be required to log in to the selected host or can be automatically logged in to the host.

Applications wanting to establish connections with applications in other hosts do not need to be aware of the topography of the network. Instead, the installation must specify the addressing and flow control parameters that the network software can use to establish the connection. This must be done for each path that can be used for an application-to-application connection.

As part of the definition of the network configuration, the installation also defines the supervision paths to hosts that an NPU can use to obtain its configuration and its operator interface.

DEFINING A CDC NETWORK

When defining your network, you must specify both its physical and its logical characteristics. The Network Definition Language helps you create your definition in a simple and logical manner. This section presents three aspects of the network that must be described in your NDL program: its hardware elements, its software elements, and the logical relationships you must define among them.

HARDWARE ELEMENTS

To ensure proper operation of the network, you must carefully define all of its hardware elements and establish the relationships among them. The configurable hardware elements of the network are shown in figure 1-10. Each type of element is described in detail below.
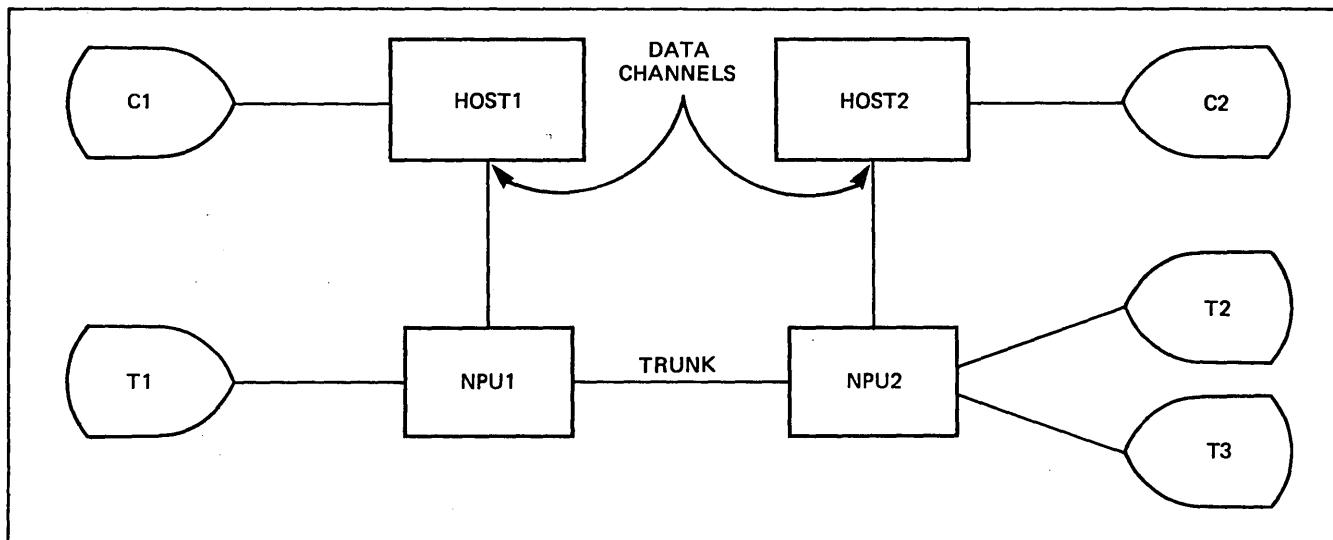
Figure 1-8. Dualnet
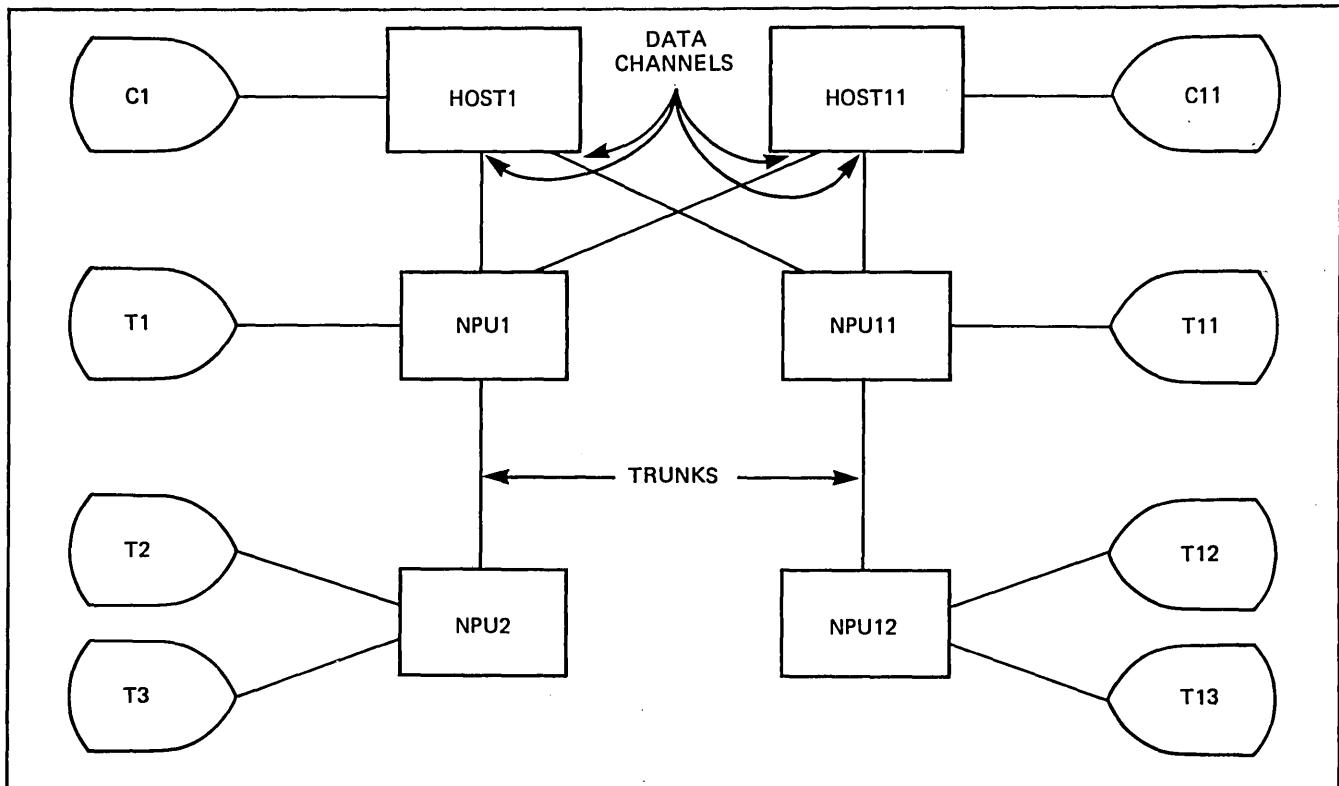
Figure 1-9. Multinet
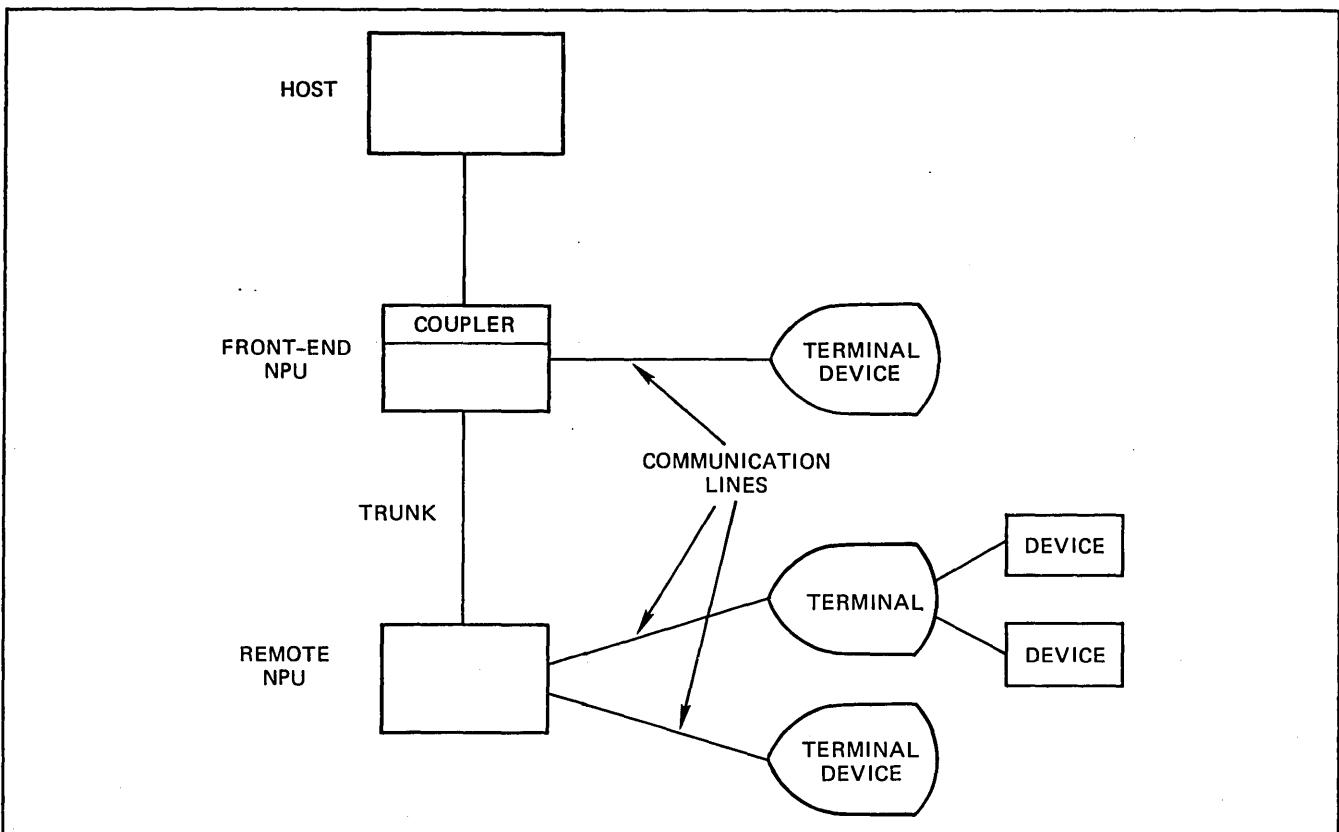
Figure 1-10. Configurable Hardware Elements

Host Processors

A host processor is any computer mainframe that is linked to the network to run network application programs. Host computers contain the portions of the network software necessary for applications to access the network.

A host can be connected through its data channels to one or more network processing unit data channel couplers, as was illustrated by the network processing units NPU1 and NPU2 of Simnet (figure 1-5), which in turn can be connected to other hosts.

No separate NDL statements exist to define a host. Hosts are assumed to be attached to couplers, so you can specify all needed host access information when you configure the couplers. Computers that access the network without couplers are treated by the network as terminals or as foreign hosts and access the network via X.25 connections. These computers must communicate through one of the terminal protocols described in sections 4 through 7.

Network Processing Units

Network processing units can be either CDC 255x series Communications Processors or non-CDC processors with compatible communication procedure software. Unless otherwise qualified, subsequent references to network processing units in this manual apply to both types of equipment.

Networks can contain two types of 255x network processing units: the 2550 (sometimes called a 2550 Host Communications Processor) and the 2551. A 2550 NPU has at least one coupler and always operates as a front-end NPU. A 2551 can have a coupler, but need not have one, and can be configured as either a front-end or remote NPU.

The 2551 series Communications Processor has both a programmable micromemory and a macromemory; each model of the 2551 series has varying amounts of macromemory, with several sizes of random access micromemory (RAM) logic. To run the current release of CCP requires 6144 words (6K) of micromemory. Because models can be modified on site, a model designation does not necessarily correspond to a specific macromemory or micromemory configuration.

Some NPUs have a magnetic tape cassette drive and an operator's console that is not part of the network. The magnetic tape cassette drive and accompanying deadman timer hardware are required if the 2551 has two couplers or is used as a remote NPU. When an NPU has a cassette drive, the drive is used to begin the loading of CCP from a host computer.

The cassette drive loads a copy of the system auto-start module program (SAM-P) from a CDC-supplied cassette. SAM-P is essentially a bootstrap loader that obtains the copy of CCP appropriate for its NPU from a copy of the Network Supervisor program in a host computer.

The cassette drive can also be used to load optional off-line hardware diagnostic software for use with the offnet NPU console.

If the NPU has an operator's console, that console is not part of the network; it connects to a special port of the NPU and is not serviced through a communications line adapter. This offnet console is associated with the NPU operator.

The offnet NPU console is used to run optional online or offline diagnostic software. If the site modifies its copy of CCP, the offnet console can be used with the internal Test Utility Package for online debugging of the code.

Both front-end and remote NPUs can be connected to terminals through the NPU's input/output ports. Terminals can be linked directly to these ports with communication lines, or indirectly via an X.25 packet-switching network.

If a network processing unit in your network is not CDC equipment, you can define it as an NPU if it runs software compatible with the internal network protocols of CDC NPUs.

Couplers

The input/output channel from the host is connected to the front-end NPU via a hardware module known as a coupler. The coupler is physically housed within an NPU but must be configured separately because some NPUs do not have couplers.

The coupler makes data signals to the host computer compatible with the hardware of the host and data signals to the NPU compatible with the NPU. An NPU can have one or two couplers, each connected to a host computer. To support a coupler, a CDC NPU must be loaded with a CCP variant that includes the Host Interface Program.

Trunks

A front-end NPU can be connected to another NPU by a communication line called a trunk. You can configure only one trunk between any pair of NPUs, but you can configure multiple trunks connecting different NPUs to any single NPU.

To support trunks, a CDC NPU must be loaded with a CCP variant that includes the Link Interface Program.

Communication Lines

The network software supports lines for synchronous terminals and for asynchronous terminals. These lines can be switched (also called dialup) or dedicated (also called hardwired). You can configure lines with speeds of up to 56000 bits per second (b/s) as one of nine general line types.

Each line type corresponds to a set of characteristics used by a CDC or non-CDC communications line adapter (CLA). There are three types of CDC communications line adapters:

Model 2560 series synchronous CLAs, used to support lines conecting mode 4 terminals, IBM 2780 or IBM 3780 bisynchronous terminals, HASP protocol terminals, or 3270 bisynchronous terminals to the NPU.

Model 2561 series asynchronous CLAs, used to support lines connecting teletypewriter-compatible or IBM 2741-compatible terminals to the NPU.

Model 2563 series high-level data link control (HDLC) CLAs, used to support trunks connecting other NPUs or lines connecting asynchronous terminals, NPUs with hosts, or foreign hosts to the NPU through X.25 packet-switching networks.

Terminals

A terminal is a single point of access between one or more devices and a communication line. The current network software provides support for five basic types of terminals:

Asynchronous terminals, such as the CDC 751, using either asynchronous protocol or X.25 protocol

Mode 4 protocol synchronous terminals, such as the CDC 200 User Terminal

Multileaving bisynchronous terminals, such as IBM Corporation's HASP workstations

Binary synchronous terminals, such as IBM Corporation's 2780 or 3780 terminals

Binary synchronous terminals such as IBM Corporation's 3270 terminal

You can configure up to three more types of terminals with a site-provided TIP for each type.

Terminals are grouped into classes by their specific hardware characteristics. A terminal class parameter is used to define some physical characteristics of the terminal and to determine the default values of operational characteristics. The network software supports 18 CDC-defined terminal classes. These terminal classes are described in greater detail in sections 4 through 6.

If you provide your own support software, you can use features of the Network Definition Language to define up to four additional terminal classes. CDC cannot anticipate the support required for such terminals, and thus cannot document such support. You can find information pertinent to defining your own terminals in sections 4 through 7.

Devices

Each terminal includes one or more separately configured devices that perform both input and output, input only, or output only. For X.25 and asynchronous protocols, a device such as a console is a terminal. For synchronous protocols, a device such as a console is only a portion of a terminal.

An interactive device such as a console can perform both input and output, and its user can participate easily in dialog with host software. A passive or batch device such as a card reader or line printer can perform only one-half of such dialog. Interactive and passive devices use different types of data structures and paths through the network.

SOFTWARE ELEMENTS

Network software runs in two environments: the host and the NPU. Standard CDC software runs in both to support many network configurations. Your site can also provide its own software to support other types of equipment and other uses. The characteristics of software not provided by CDC are unknown, and therefore are not addressed by this manual.

Data Structures and Flow

Data flow in the network is defined from the viewpoint of the host computer. Data coming to the host is said to be traveling upline; data moving away from the host is said to be traveling downline. This concept is shown in figure 1-11.
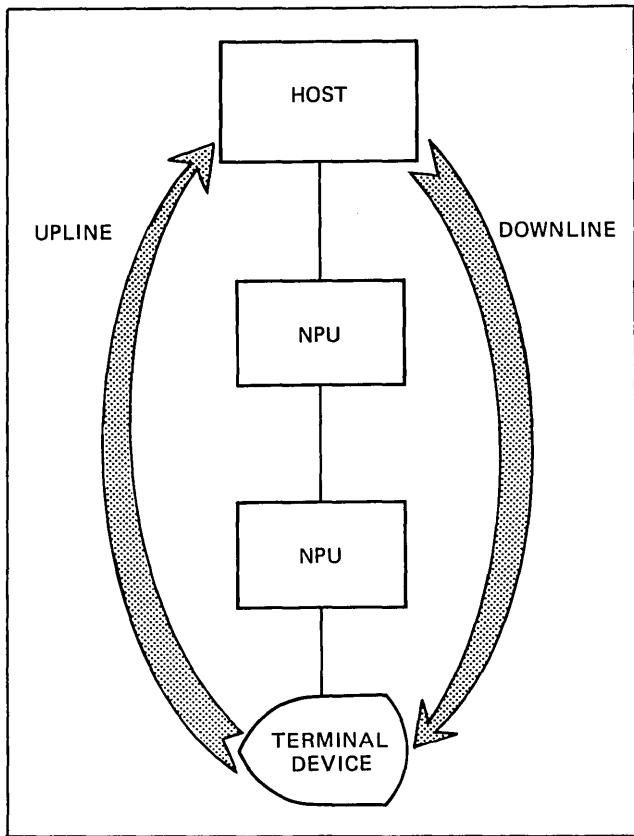
Figure 1-11. Data Flow Directions

Upline and downline information within the host and NPUs is always grouped into physical network blocks. Network data blocks are grouped into logical messages. Messages exchanged between an NPU and a device can also be grouped into physical transmission blocks of one or more logical lines. Figure 1-12 shows these concepts.

A single network data block is a collection of character bytes, analogous to a clause in English. It is a partially independent unit of information and might need to be used with other blocks to form a message.
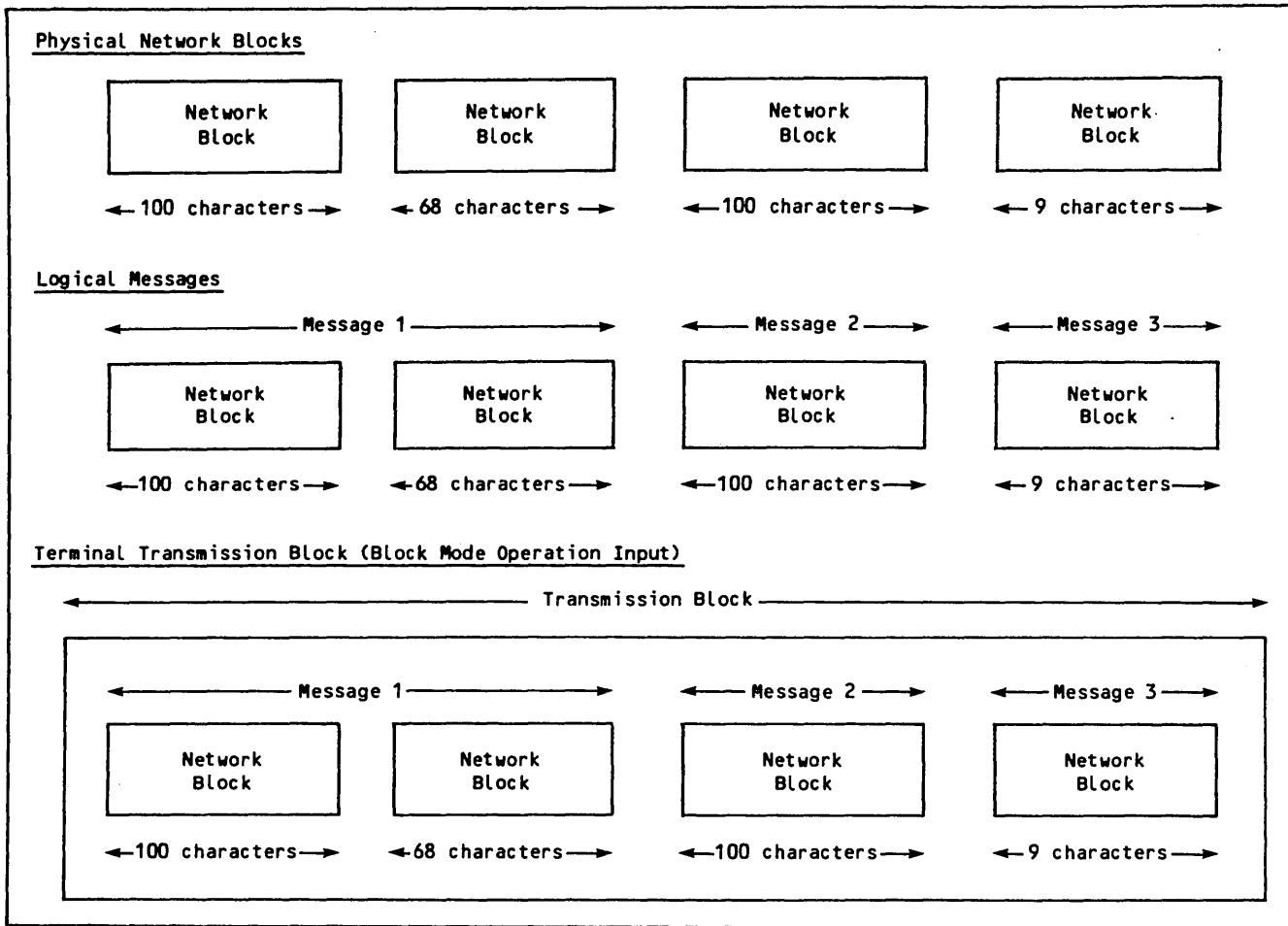
Figure 1-12. Physical and Logical Information Structures

A network data block can contain all or part of a message; you control which occurs by establishing the size a network data block can have. The size of an upline network block is fixed at the number of characters you specify when you configure the device that produces it. The size of a downline block is chosen by the application program sending it, but you can recommend a value to that program when you configure the device that receives it. You use the UBZ and DBZ parameters to control these sizes; these parameters are described in sections 4 through 7.

In upline data, a logical line of information is a single message, analogous to an English sentence. It is a complete unit.

Upline logical lines end when a device transmits certain characters or causes certain events. An upline message is normally transmitted to the host as soon as a logical line ends. A logical line ends when CCP receives an end-of-line indicator (the end of a logical line).

Terminals can transmit one line at a time to the NPU and wait for output or they can transmit several lines at a time (block mode). With line-by-line transmission, CCP detects the end of each logical line by an end-of-line character or indicator. Each logical line becomes a separate network message when the NPU receives it. Block mode terminals group logical lines in the terminal until the transmission key is pressed; these groups reach the network software as a single transmission block. The network software forwards each message to the application as a separate transmission. With block mode transmission, CCP detects the end of a transmission block by an end-of-block character or indicator.

For upline data, you can configure a device so that CCP ignores the character or event that normally causes it to detect the end of a logical line and transmit a message to the host. Instead, you make CCP use a different character or event to trigger transmission to the host. This option allows the terminal user to pack many logical lines into one set of upline network blocks; each line includes the character that terminates it. The host receives only one message. From the application and the terminal user's viewpoint, one message is many logical lines. This mode of operation is suitable mainly for block mode terminals where the user can compose an entire screen of data and then transmit the entire screen to the NPU at one time.

If a single message fits into an upline network data block (you should try to make this happen as often as possible), the network data block becomes a single-message block. If one upline message cannot be fit into a network data block, CCP splits it into as many blocks as necessary. The text of sections 4 through 7 assumes that each network data block contains a complete message.

A downline message ends when an application program indicates that condition. A downline message can contain many logical lines of output.

Network data blocks are restructured into other types of blocks at points of entrance and exit from the network processing units. Figure 1-13 shows these points as circles. Transmission blocks exchanged with X.25 devices are called packets and have different size and protocol content requirements than transmission blocks exchanged directly with a terminal. You can control some of the characteristics of packets.
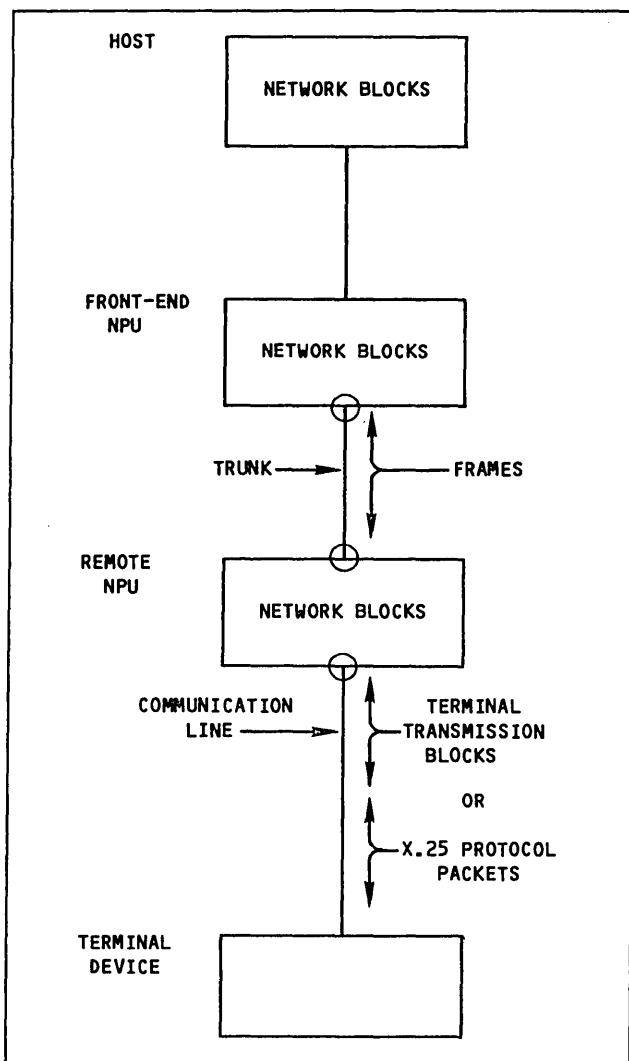
Figure 1-13. Block Reassembly Points

Data blocking and temporary storage (queuing) occur for upline and downline traffic at several points in the network. You control the storage required by controlling the network block size and the number of blocks queued in each direction. You do this by declaring values for the upline and downline block limit, for application block limit, and for upline and downline block size (UBL, DBL, ABL, UBZ, and DBZ) parameters of each device.

Network Data Blocking for Interactive Device Input

A network data block is created every time the number of characters transmitted from a device exceeds the upline block size. A message block is created each time the terminal user enters an end-of-line character. You define the character that must be entered from a device in the network to end a logical line.

A block transmission exists each time a group of logical lines is transmitted as a unit by a device. If the terminal user enters an end-of-block character, any incomplete logical line is terminated and a message block is generated. You define the character that must be entered from a device in the network to end a multiple-message block.

Network Data Blocking for Batch Device Input or Output

Batch devices require large amounts of data to be moved to or from a host computer's mass storage devices. Batch upline data is therefore assembled into messages of multiple network data blocks. Each network data block contains one or more mass storage physical record units (PRUs). Batch downline data is disassembled from such sets of blocks.

Terminal Output Data Blocking

Terminals sometimes require data to be received in blocks of a maximum length, independent of the data's structure within the network. Such blocks are called transmission blocks; you can define the transmission block size for each device in the network using its XBZ parameter.

Link Data Blocking

Transmission blocks exchanged between NPUs have different structure and content requirements than those exchanged directly with terminals. Such physical link transmission blocks are called frames. Regardless of data type, network data blocks are reassembled into frames of a maximum number of bytes before transmitting them across a trunk. Data received from a trunk is reassembled into network data blocks. Frames are constructed and disassembled at points also shown with circles in figure 1-13. You must define the number of character bytes that comprise a frame on each trunk in the network.

CDCNET data structures are discussed in the CDCNET Terminal Interface Usage Manual.

Host-Resident Software

Two types of programs are associated with the host computer: CDC-written application programs and site-written application programs.

CDC-Written Application Programs

CDC host software includes programs that support a number of network device applications. The following CDC-written programs are currently supported:

The Interactive Facility (IAF)

The Remote Batch Facility (RBF)

The Transaction Facility (TAF)

The Terminal Verification Facility (TVF)

The Message Control System (MCS)

The PLATO/NAM Interface-PNI (PLATO)

The Printer Support Utility (PSU)

The Permanent File Transfer Initiator (PTFI)

The Permanent File Transfer Server (PTFS)

The Queue File Transfer Initiator (QTFI)

The Queue File Transfer Server (QTFS)

The NOS/VE Interactive Facility (VEIAF)

The Interactive Transfer Facility (ITF)

The Network Log Server (NETLS)

The Network Operator Facility (NOF)

The Network File Server (NETFS)

The Initialize MDI Server (INITMDI)

The Network Log Termination Utility (NLTERM)

IAF provides the terminal user with the ability to create files and programs from an interactive device and to execute jobs from that device.

RBF transfers job input files between remotely located batch terminals and a host computer. It also allows monitoring of a job's progress through the network and provides file output at the remote batch terminal.

TAF provides on-line transaction processing, allowing the terminal user to manipulate information in a data base directly from the terminal. TAF requires very little dialog to update a data base.

TVF provides the terminal user with the ability to test an interactive device's data transmitting and receiving functions.

MCS provides the terminal user with a mechanism for transferring data between an interactive device and a COBOL program or between two COBOL programs.

PNI provides the CDC 721 Viking Extended terminal user with the ability to log into the PLATO lesson delivery and authoring software in normal asynchronous mode. The PLATO software then loads its special software into the 721 so that the operator can execute lessons or use the other facilities of PLATO software.

PSU provides the ability to print NOS queue files on a C533/C536 printer using asynchronous connections to the NPU.

PTFx provides a mechanism for using NAM and NPUs to transfer permanent files between hosts.

QTFx provides a mechanism for using NAM and NPUs to transfer queue files between hosts.

VEIAF provides the terminal user with the ability to create files and execute jobs on NOS/VE.

ITF provides the terminal user with the ability to connect to the CYBER 200 family of computers.

NETLS provides logging functions for the CDCNET network.

NOF provides operator interface functions for the CDCNET network.

NETFS provides file access functions for the CDCNET network.

INITMDI provides dump and load MDI functions for the CDCNET network.

NLTERM provides network log termination functions for the CDCNET network.

Site-Written Application Programs

If your installation runs its own application programs, characteristics of each application program might need to be relayed to the network software.

For more information on writing network application programs, see the Network Access Method Version 1 Host Application Programming Reference Manual listed in the preface.

NPU-Resident Software

Each CDC NPU in the network contains a copy of the Communications Control Program (CCP) tailored to the specific functions to be provided by that NPU. Each NPU might require its own variant of CCP, depending on the macromemory size of the unit and the software modules that need to reside in the unit. Variants are generated as described in the NOS Version 2 Installation Handbook. These variants reside in the host NPU load file (NLF). The appropriate variant for loading into each NPU is determined by the host Network Supervisor program from information contained in the network configuration file.

If the network processing units in your network are CDC 255x Communication Processors, you will probably be concerned with only the interface programs. These are the HIP, LIP, and TIP modules depicted in figure 1-14.
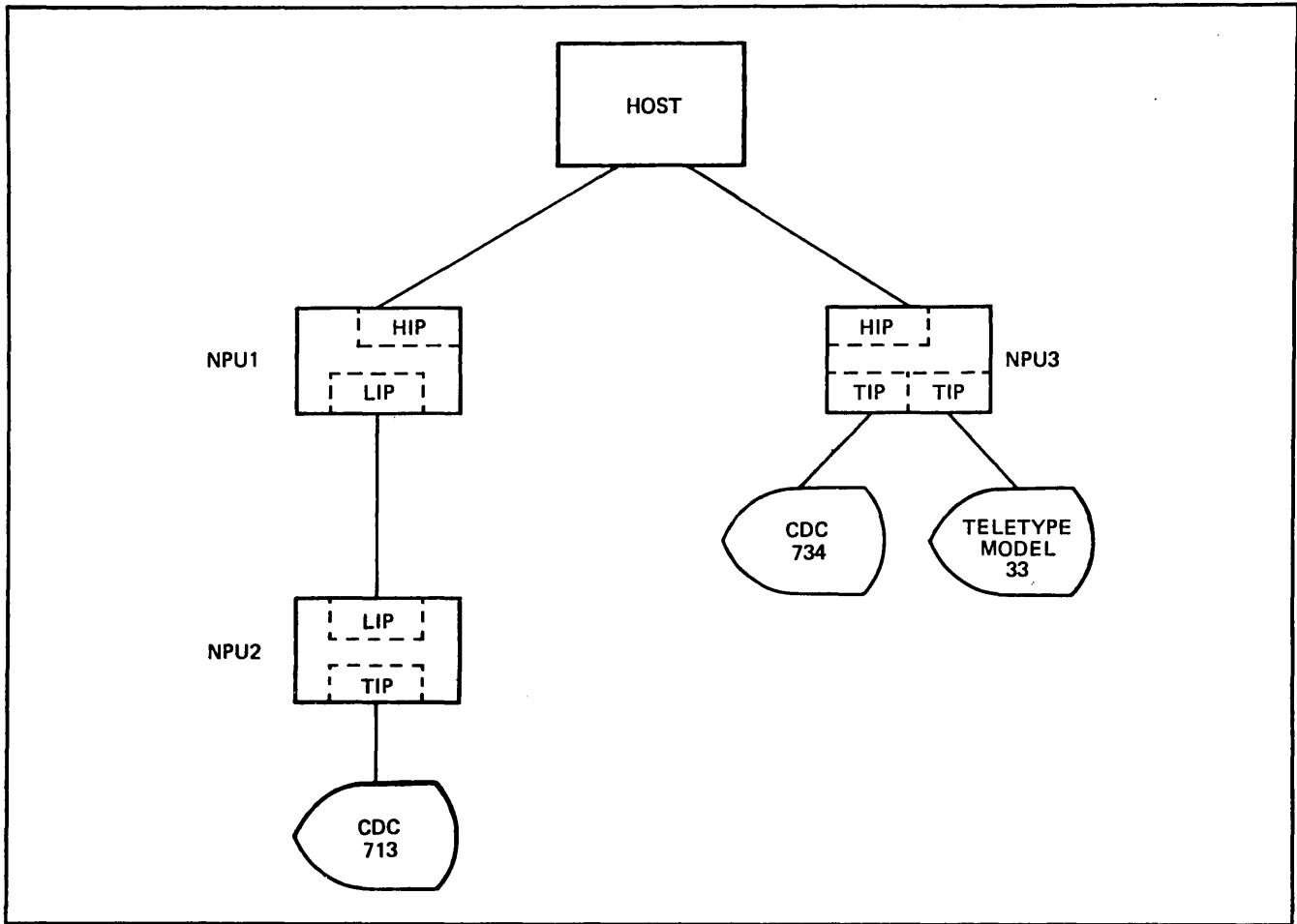
Figure 1-14. CCP Interface Program Configuration

Your concern with the HIP and LIP modules is likely to be limited to ensuring that the CCP load variant for each NPU contains only the necessary modules. For example, NPU2 is a remote NPU, so it does not need a HIP. Since NPU3 does not service a remote NPU, it does not require a LIP. Since a LIP requires additional memory, you would want a CCP variant for NPU3 that does not include a LIP.

The TIPs contained in each CCP load variant must correspond to the TIPs referenced in the LINE, GROUP, TERMINAL, and TERMDEV statements for the NPUs into which the variant will be loaded. For example, in figure 1-14, because NPU2 only supports CDC 713 terminals, it need only contain the ASYNC TIP. NPU3, however, must contain both the MODE4 TIP for the CDC 734 and the ASYNC TIP for the M33 Teletype.

For more detailed information about CCP, refer to the Network Access Method Version 1/Host Application Programming Reference Manual listed in the preface.

CDCNET Network

CDCNET is Control Data's method for implementinng distributed communications. Figure 1-15 shows the basic elements of a CDCNET network discussed below.

CDCNET Device Interfaces

Small communications processors called device interfaces (DIs) constitute the hardware portion of CDCNET.

Because CDCNET distributes major communications functions throughout a network, DIs perform different functions depending on their particular network task:

The Mainframe Device Interface (MDI) lets you connect a host CYBER computer system to a local area network.

The Terminal Device Interface (TDI) lets you connect user terminals and workstations to a local area network.

The Network Device Interface (NDI) lets you connect one CDCNET local area network to other networks.

The Mainframe Terminal Interface (MTI) lets you connect user terminals/workstations to a CYBER host without requiring that they be tied into a local area network.

You can find further information on CDCNET network device interfaces in the CDCNET Reference Manual.
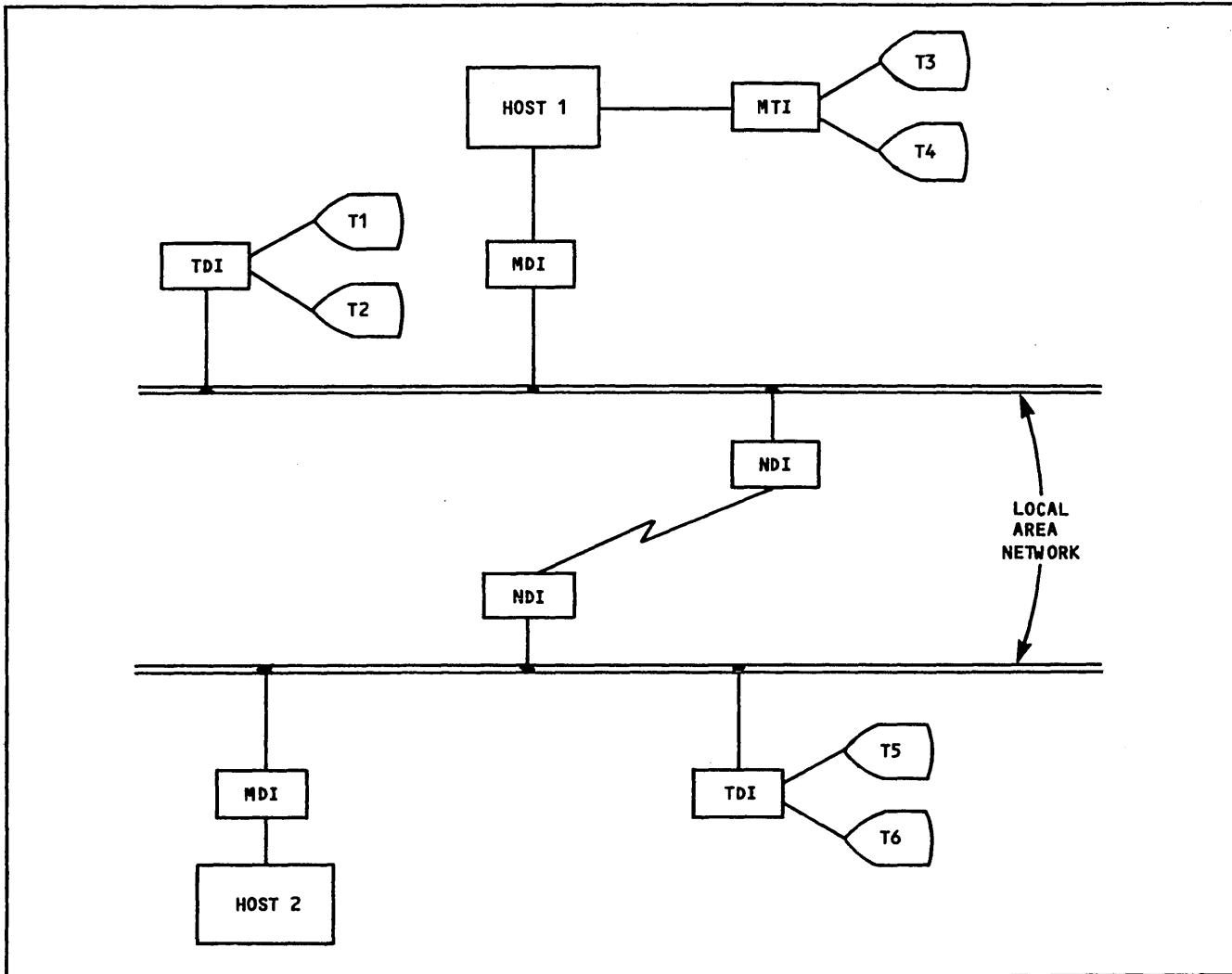
Figure 1-15 CDCNET Networks

CDCNET Device Interface Software

The software which resides in a device interface is comprised of the following:

Base system software

Layer software

Interface software and gateways

Network management software

Base System Software

The base system software performs two major tasks:

Initializes the operation of the device interface (DI)

Maintains the operational environment of the DI by serving as its executive routine, detecting and reporting DI malfunctions, and managing the DI's onboard diagnostics

You can find more information on the CDCNET base system software by referring to the CDCNET Systems Programmer's Reference Manual, Volume 1, Base System Software.

Layer Software

CDCNET layer software enables applications software, end users, terminals or workstations, and host computers to exchange information through a compatible set of protocols and interfaces.

You can find more information on the CDCNET layer software by referring to the CDCNET Systems Programmer's Reference Manual, Volume 2, Network Management Entities and Layer Interfaces.

Interface Software and Gateways

CDCNET interface software and gateways consist of various software packages that enable CDCNET DIs and hosts that accomodate Control Data Network Architecture (CDNA) to communicate with other

hosts, networks, and terminals/workstations which do not support CDNA.

You can find more information on the CDCNET interface software and gateways by referring to Volume 2, Network Management Entities and Layer Interfaces and Volume 3, Network Protocols of the CDCNET Systems Programmer's Reference Manual.

Network Management Software

CDCNET network management software performs daily and periodic tasks related to the administration, maintenance, and operation of the communications network.

You can find more information on the CDCNET network management software by referring to the CDCNET Systems Programmer's Reference Manual, Volume 2, Network Management Entities and Layer Interfaces.

LOGICAL ELEMENTS

In addition to the hardware and software elements of your network, you must define certain logical elements. These logical elements establish the

association between terminals and application programs, as well as between one application program and another. Using these logical elements, the network software performs as though it were independent of the physical organization of the network.

Nodes and Logical Links

A logical path concept is used to describe the connections between elements of the network in a manner that is independent of the physical data paths between them. Each junction of possible branches in these logical paths must be known to the network. These junctions are called network nodes. In your NDL program, you identify each network node by assigning the hardware that contains it a unique node number from 1 to 255. The network software uses these numbers to route messages from a source node to a destination node.

Figure 1-16 shows the nodes (node numbers are circled) in a multiple-host network. Node number 1 is assigned to the coupler. Because the coupler is the means of addressing the host, the coupler is known as the host node.
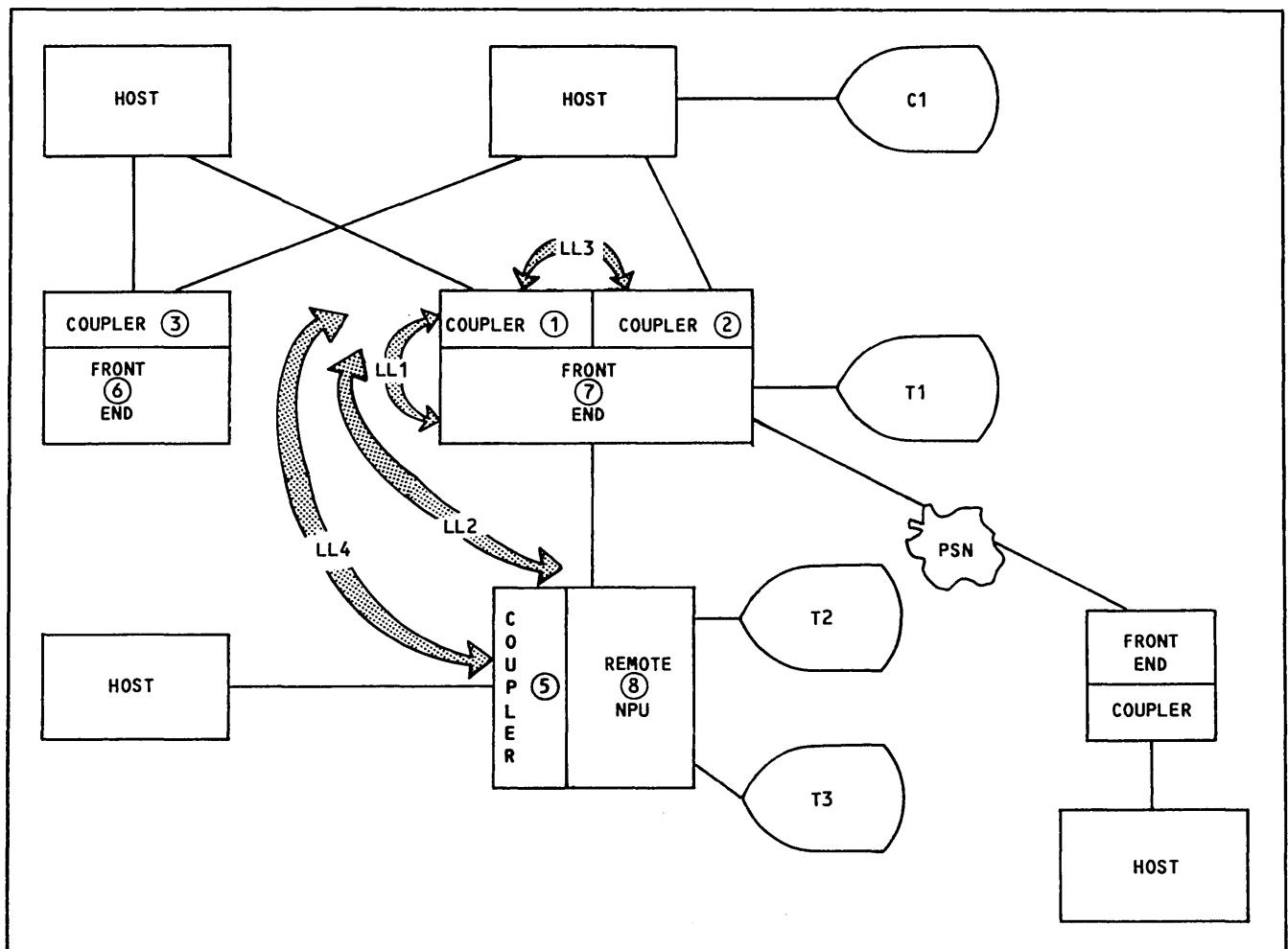
Figure 1-16. Nodes and Logical Links

The front-end NPU has a separate node from its coupler because a front-end NPU must be separately addressable from its couplers. Node 7 and node 8, within the front-end and remote NPUs, are known as terminal nodes.

Figure 1-16 shows the four logical links in the network, labeled LL1 through LL4, originating in coupler 1. Logical links are known by the end points of the path they traverse. One end point is always a host node in a coupler; the second end point is a terminal node in an NPU or another host node.

You must define each logical link that originates in a coupler when you configure the NPU that houses the coupler. You do so by identifying the NPU that is the terminal node or other coupler node of the link.

Table space for node numbers is reserved within the host software, with space assigned to all numbers from 1 to the highest number used; unused numbers have corresponding table space. You can minimize memory resource requirements by assigning node numbers sequentially, giving host nodes lower values than terminal nodes.

You must specify for each NPU the logical links to hosts which are allowed to configure and control the NPU. These are called supervisory links. When each trunk between NPUs is defined, you specify whether either NPU can be loaded via the trunk. Loading, configuration, and control are supervisory functions which can take place only between hosts and NPUs that belong to the same site.

Subnetworks or Sites

A site is defined as being all of the hosts and NPUs under single administrative control. NPUs can be loaded and supervised only by hosts within their site, but terminals connected to an NPU and applications running in a host can connect to hosts belonging to different sites.

When you define a network, you include all of the NPUs in your site as well as all of the NPUs belonging to other sites to which your site's NPUs have physical and logical connections. Lines and terminal definitions need to be specified only for the NPUs belonging to your site, and the supervisory links specified for each NPU must all be to hosts that belong to the NPU's site.

The concept of a site is very important when upgrading release levels of NPU and host software. Throughout the remainder of this manual, except where specifically stated, the term network will refer to a single-site network.

Logical Configuration of Hosts

You can describe a host computer in your program in terms of the application programs it is to run. A local division is provided in the NDL program structure for this purpose.

Logical Configuration of Terminals

In addition to the hardware description of your terminals specified in the network division, you

can supply, in the local division, information concerning their ability to access network resources. This information includes:

Mandatory, primary, or default values for login operating system validation parameters

Mandatory or primary default values for login application program name

APPLICATION-TO-APPLICATION CONNECTIONS

Figure 1-17 shows an application-to-application connection within the same host. Figure 1-18 shows that interhost application-to-application connections can either be made via a trunk or a single NPU. Figure 1-19 shows the possibilities for application-to-application connections via a X.25 packet-switching network.
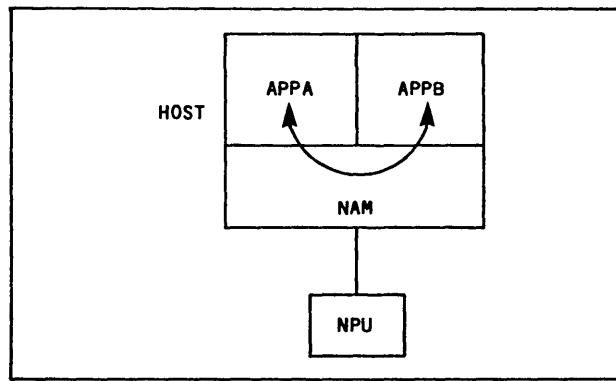
Figure 1-17. Intrahost Application-to-Application Connection

The paths taken by application-to-application connections are described by the INCALL and OUTCALL statements. These statements are a part of the local configuration file. They are used by NVF to indicate which application programs are allowed to request connections between applications and to describe the possible paths for connections between applications.

You use the OUTCALL statement to indicate an application program that can initiate a connection to another application program and a path for reaching the application from the host that you are defining. To initiate a connection to another application, an application references an OUTCALL statement in its request.

You can also use the OUTCALL statement to indicate particular paths in the network for which NAM should perform dynamic updating of the LID/PID table in the operating system. This table is accessible to both the NOS console operator (refer to the NOS analysis handbook), as well as end users of the NOS operating system (refer to the NOS reference set, Volume 3, System Commands).

You use the INCALL statement to specify the possible paths in the network to an application program of the host being defined on which application-to-application connections can be received. The application specified in the INCALL statement is
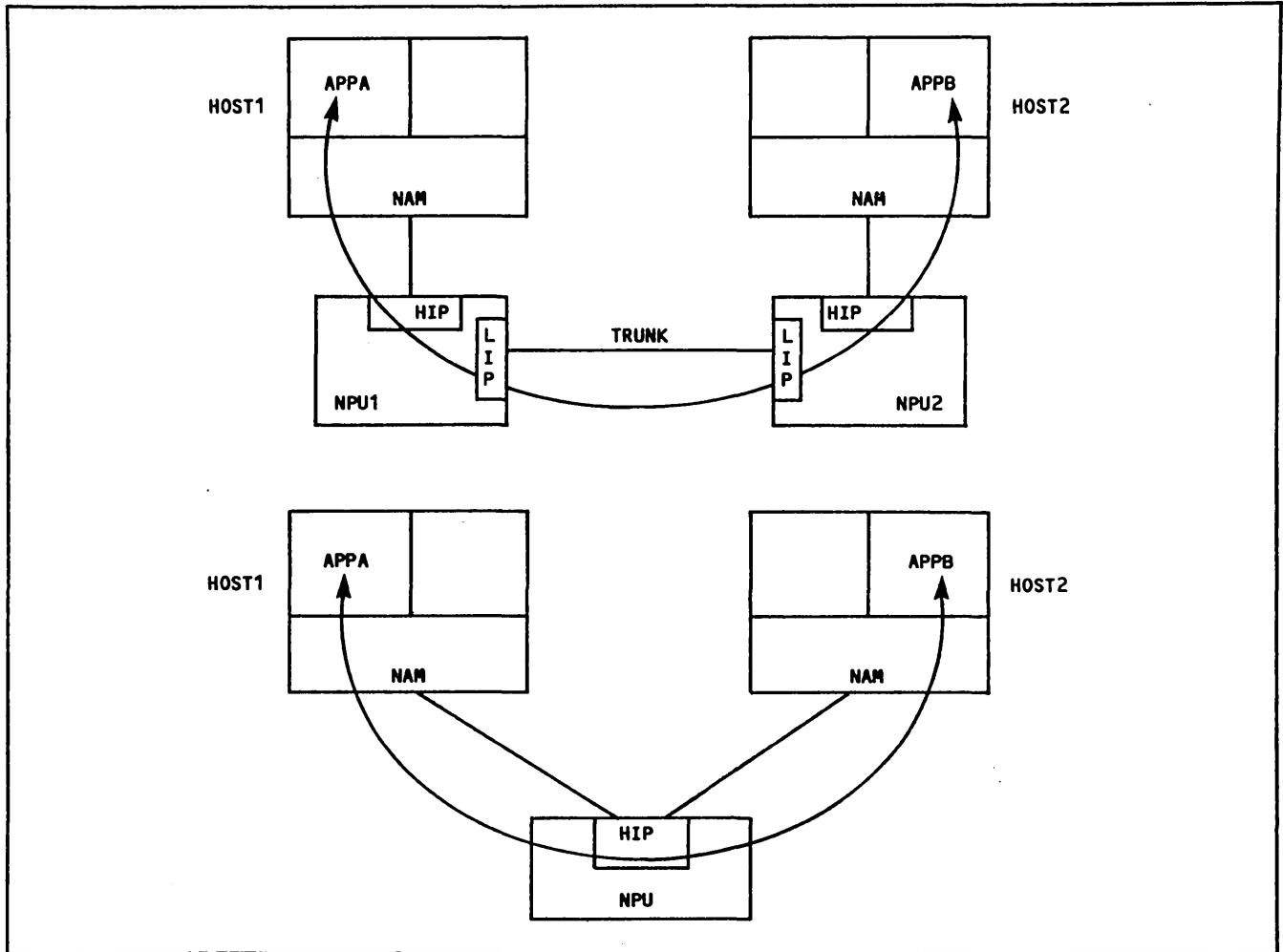
Figure 1-18. Interhost Application-to-Application Connections

requested to accept a connection from another application.

MULTILEVEL SECURITY

Access to files by terminal users can be limited by the access level (AL parameter) assigned to the communication line by which they access the network.

NOS multilevel security is based on the use of hierarchical security access levels. There are eight security access levels available for the classification of data, ranging from level 0 to level 7. The lowest (unsecured or unclassified) level is 0, and the highest (most secured or classified) level is 7.

The creator (or owner) of a permanent file, a local file, or a magnetic tape can assign an access level to that file, depending upon individual validation privileges and upon operating system constraints. Alternatively, the owner can allow the access level of the file to be assigned automatically by the operating system.
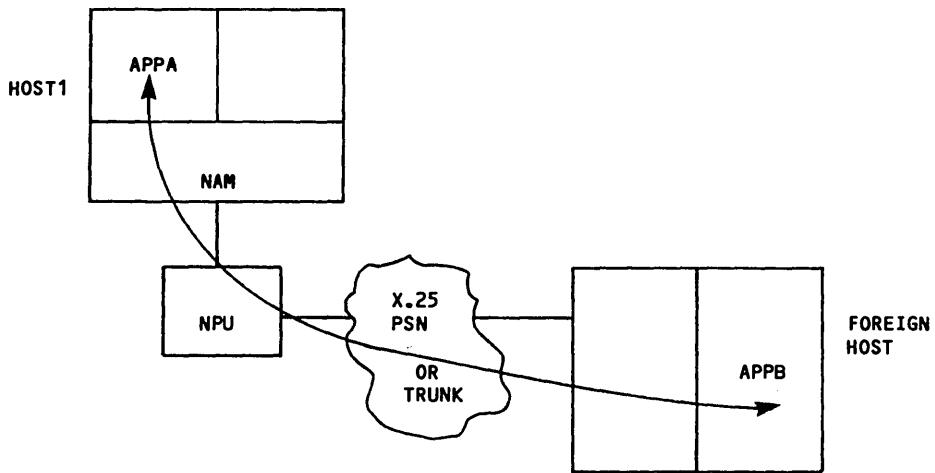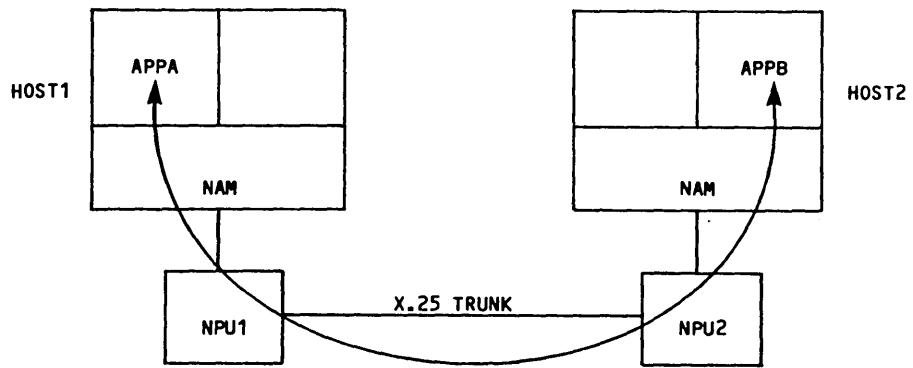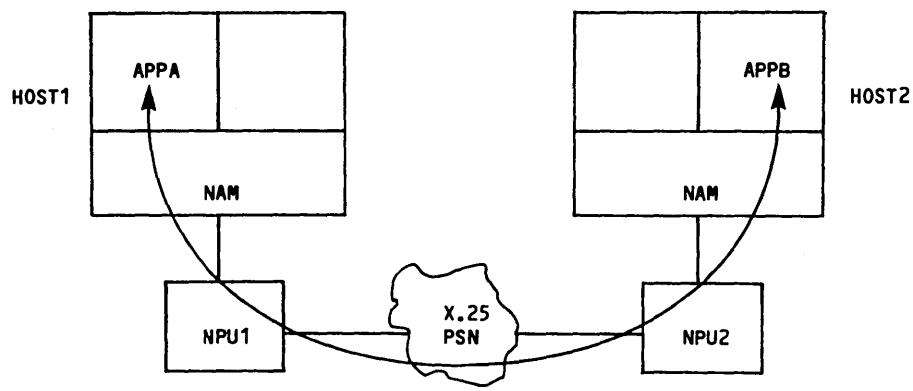
Figure 1-19. Interhost X.25 Application-to-Application Connections

This section presents the format and functional syntax of the Network Definition Language. The NDL program statements you will use to create the network definition files are described in separate subsections.

All the examples in this section and sections 3 through 8 refer to the hypothetical network SVLnet, shown in figure 2-1.

FUNCTIONAL SYNTAX AND FORMATS

Each NDL statement has the general form given in figure 2-2. Separators, terminators, and format representation conventions are explained on the notation page at the front of this manual. The formal syntax of the Network Definition Language (in Backus-Naur form) and a concise summary of each statement's syntax and format are given in appendix E.

Unless otherwise indicated in the format descriptions presented later in this section, all values declared for keywords are either unsigned decimal integers or character strings of one through seven letters and digits. Element names are also restricted to one initial letter and up to six more letters and/or digits. All value declarations identified as hexadecimal integers are also unsigned. NOS user names declared as values can contain asterisks.

VALUE DECLARATION FORMATS

Value declarations are position-independent. You can place declarations within a statement in any order. You do not use commas to indicate omitted declarations; such null-value parameter declarations cause an error diagnostic.

All value declarations can have the form:

   keyword=value

All optional keyword=value declarations have an explicit value that is equivalent to omitting the declaration.

You can specify any keyword more than once in a single statement; the latest declaration in the statement takes precedence. This ability allows you to override previously established values.

For example, a DEFINE statement can contain a declaration for the DLX parameter that is used in the definition of all asynchronous consoles of the network. That parameter can be reset to a different value for one terminal by placing a specific DLX keyword declaration after the DEFINE statement identifier on the appropriate DEVICE or TERMDEV statement.

The format you use to declare parameter values depends upon the parameter being declared and upon the presence or absence of the value. The value portion of the declaration can have one of two forms:

   value-required

   stand-alone

Value-Required Keywords

A declaration containing a value-required keyword has the form

   keyword=value

For example, the following statement contains a value-required keyword (NODE) and an explicit value (1) in its declaration:

   CPLR1: COUPLER NODE=1.

You can use either explicit or implicit values for value-required keyword declarations. The legal values for explicit declarations are given in sections 3 through 7. Implicit value declarations are allowed only for automatic-recognition dependent, CCP-default, or no-default keywords; after such a declaration, the network software implies the value to use from the context of the parameter.

Automatic-Recognition-Dependent Keywords

The automatic-recognition-dependent keywords are:

   CA   STREAM
   CO   TA
   CSET   TSPEED
   STIP

If you declare an explicit value for one of these keywords, the declared value is always used. You can also declare a value of AUTOREC for such keywords and use the implicit value for the parameter.

When you use the value AUTOREC, CCP determines the value needed when it performs automatic recognition for the communication line or terminal. (AUTOREC can be used only if the line is an automatic recognition line.)

Using the value of AUTOREC has the same effect as not specifying the keyword. Specifying AUTOREC makes your network definition less dependent on terminal type.
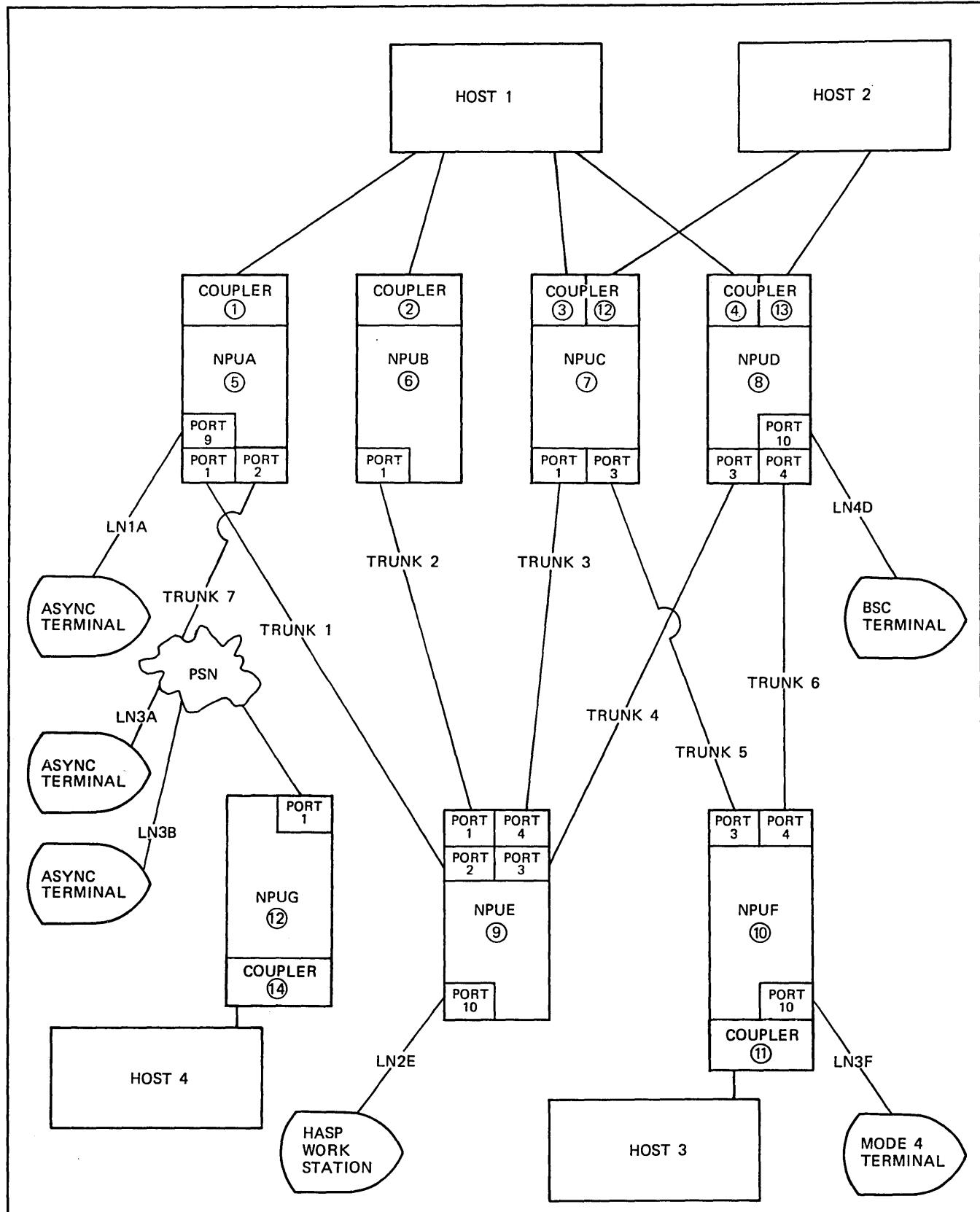
Figure 2-1. SVLnet - Configuration Example

$$[\text{ element-name:] } [\Delta] \text{ statement-name } \Delta \left[\begin{Bmatrix} \text{keyword1=value1} \\ \text{keyword1} \end{Bmatrix} ,\ldots, \begin{Bmatrix} \text{keywordn=valuen} \\ \text{keywordn} \end{Bmatrix} \right] . [*\text{text}]$$

element-name   The name given to the entity being defined. When the NDL processor requires an element name, the name specified must be unique within a division. Element names comprise one alphabetic character optionally followed by up to six alphabetic or numeric characters from the 6-bit display code set. The host and NPU operators use the element name to monitor and alter the status of the named entity.

:   The colon separates the element name from the reserved statement name. When an element name is specified, a colon must follow it. The colon does not become part of the element name.

Δ   The Greek capital delta character represents a separator. Blanks and commas are separators and are required where shown in the prototype statement formats; blanks can be used where commas are shown. One blank is sufficient as a separator, and multiple blanks are treated as one. One or more blanks preceding or following a colon or comma are ignored.

statement-name   A reserved word specifying the statement type. Each statement type has a unique statement name.

keywordi=valuei or keywordi   A value declaration of the statement. This has two possible formats, depending upon the keyword and upon inclusion of the value in the declaration. (See Value Declaration Formats.)

,   Commas (and blanks) are separators. Separators are required between declarations. A comma preceded or followed by one or more blanks is treated as a single separator. A comma indicates that the NDL processor should interpret whatever follows it as a new keyword.

.   A period is required to terminate each statement.

*text   Comment text can appear on the same line as a statement, but must be written after the period, preceded by an asterisk, and cannot be continued on the next line without specifying COMMENT or * again.

Figure 2-2. Functional Syntax of NDL Statements

CCP-Default Keywords

The CCP-default keywords are:

AB	CT	ELR	OC	XLC
BR	DLC	ELX	OP	XLTO
BS	DLTO	EP	PA	XLX
B1	DLX	IC	PG	XLY
B2	EBO	IN	PL	
CI	EBR	LI	PW	
CN	EBX	LK	SDT	
CP	ELO	MREC	TC	

If you declare an explicit value for one of these keywords, the declared value is always used. You can also declare a value of CCP for such keywords and use the implicit value for the parameter.

When you use the value CCP, the Communication Control Program (CCP) supplies the default value for the terminal class when it configures the device. (CCP should be specified only if the line uses a TIPTYPE value for a CDC-written Terminal Interface Program.) Using the value CCP has the same effect as not specifying the keyword except that it overrides a previously declared explicit value.

No-Default Keywords

The no-default keywords are:

AL	MFAM
DFAM	MUSER
DUSER	PAPPL
HN	PFAM
MAPPL	PUSER

If you declare an explicit value for one of these keywords, the declared value is always used. You can also declare a value of NONE for such keywords.

When you use the value NONE, the NDL processor ignores the keyword declaration. Using the value of NONE has the same effect as not specifying the keyword.

Specifying NONE allows you to override a previously declared explicit definition for the parameter value.

Stand-Alone Keywords

A declaration containing a stand-alone keyword can have the same form as a declaration containing a value-required keyword, or it can have the form

    keyword

For example, the following statement contains a stand-alone keyword (DI) as its second declaration:

    LGLK1: LOGLINK NCNAME=NPUA,DI.

The stand-alone keywords are:

ARSPEED	EOF	PRI
AUTO	HD	PRIV
AUTOCON	IMDISC	PRU
BCF	KDSP	RIC
COLLECT	NETXFR	RS
DCE	NOLOAD1	UID
DI	NOLOAD2	XAUTO
DMP	OPGO	

You can use either explicit or implicit values for these keyword declarations. If you declare an explicit value for one of these keywords, the declared value is always used. You can also omit such keywords and use the implicit value as a default for the parameter.

The legal values for explicit declarations are YES and NO. An omitted stand-alone keyword usually has an implicit value of NO as its default value. Specifying the keyword alone is the same as specifying the explicit value of YES. If you want to change a previously declared explicit value of YES or document the use of the default value, you can use the explicit value of NO. If you want to change a previously declared stand-alone keyword, you can use the explicit value of NO.

For example, consider the following statement:

    LGLK1: LOGLINK NCNAME=NPUA,DI=NO.

This statement has an effect opposite to the first example given. The NO value says that the logical link is not disabled (it is enabled).

PARAMETER VALUES

Parameter values can be specified either as an unbroken sequence of alphabetic and numeric characters or in string form enclosed in single quotes.

String parameter values can span one or more lines and can contain embedded blanks for readability. Only alphabetic or numeric characters are taken from string parameters. Embedded blanks are ignored, and the parameter is terminated by any nonblank, nonalphabetic or nonnumeric character (preferably a single quote).

For example, either of the following formats (which result in identical definitions) can be used:

    PAD = 1A2B3C4D5E6F2A3B4C5D6E7F.
or
    PAD = '1A2B 3C4D 5E6F
          2A3B 4C5D 6E7F'.

STATEMENT LENGTH AND CONTINUATIONS

NDL statements are assumed to be input from either card readers or terminal keyboards. Because these input sources are physically different, there are no restrictions (such as card column or line character position usage) on the format of the statements. You can define your own formatting conventions, with the following exceptions:

> You must divide NDL statements into units of 90 or fewer characters. Characters 73 through 90 of each unit are reserved for your convenience as an Update or Modify sequence number field and are not interpreted by the NDL processor. Characters in the sequence number field, however, are included in the count of characters in each statement unit.

> If you split a statement of more than 72 meaningful characters into two or more units, the meaningful portion of each unit must end with a legal separator.

> You cannot divide a value declaration over two units.

> You must terminate all statements except COMMENT statements with a period. If you do not terminate a statement with a period, NDLP assumes that the next statement is a continuation of the present one; this assumption can produce errors. Comment text can follow a period if the text is preceded by an asterisk.

> You cannot declare more than one statement in a single unit.

> You cannot continue comments on the next line unless COMMENT is specified at the beginning of each comment line; that is, you cannot break COMMENT statement text into separate units within a single statement.

> There is no limit on the number of units into which you can break a statement. Empty (blank) units are permitted.

These constraints are illustrated by the examples in figure 2-3.

DEFINITION STRUCTURE

The structure of the NDL program that defines your network will parallel the physical and logical structure of the network. An NDL program can contain two formal types of divisions:

> A network division, corresponding to each network configuration file (NCF) to be produced

> A local division, corresponding to each local configuration file (LCF) to be produced

To configure your network initially, you must provide a NCF for each host in the network running the Network Supervisor (NS) or the Communications Supervisor (CS). The NCFs do not need to be exactly the same. You must also provide a LCF for each host in the network for the Network Validation Facility (NVF). All hosts may use the same LCF. The NCF and LCF can be created with a single NDL program containing two or more divisions or with

In these examples, the character sequence of Δ...Δ represents an ellipsis of blanks and does not contain a period as part of the represented input.

The following three-unit statement is correct:

    GOODTRY: ΔCOUPLERΔNODE=1, Δ...Δ
    Δ...Δ
    LOC=PRIMARY.

The following two-unit statement is incorrect because it divides a value declaration with a separator (a blank):

    BADTRY1: ΔCOUPLERΔNODE= Δ...Δ
    1, ΔLOC=PRIMARY.

The following two-unit statement is incorrect because it omits the period terminator:

    BADTRY2: ΔCOUPLERΔNODE=3, Δ...Δ
    LOC=PRIMARYΔ

The following one-unit statement is incorrect because it contains more than 72 characters and is not divided into two units at a separator:

    DEVICE1: ΔDEVICEΔDT=DT12, ΔABL=7, ΔDBZ=2043, ΔUBZ=2000, ΔDBL=7, ΔUBL=31, ΔXBZ=2043, ΔPRI.

Figure 2-3. Statement Continuation Examples

two or more NDL programs that contain one division each. Subsequently, you can update the network configuration by changing either the NCF or LCF or both.

DIVISION HIERARCHY

An NDL input file can consist of several network and local divisions; these divisions can appear in any order within the network definition. NDLP creates one network configuration file for each network division it encounters while processing your program. NDLP also creates one local configuration file for each local division in your program.

STATEMENT HIERARCHY

The relationship between network elements is defined by the placement of the statements used to define them. For example, a COUPLER statement that follows an NPU statement defines a coupler that is connected to that NPU. Further, a LOGLINK statement that follows the COUPLER statement defines a logical link that originates in that coupler.

Statements must occur within the proper division and, except for the following statements, must occur in the order shown in figure 2-4:

    APPL   OUTCALL
    COMMENT   TITLE
    DEFINE   TRUNK
    INCALL   USER

SPECIAL-PURPOSE STATEMENTS

Four special-purpose statements are supplied to document, simplify, and separate network definitions:

    TITLE statement

    COMMENT statement

    DEFINE statement

    END statement

The DEFINE and COMMENT statements are not shown in figure 2-4 because they need not appear in a particular order or within a particular division.

TITLE STATEMENT

A TITLE statement allows you to title each page of your output listing. The format of the TITLE statement is shown in figure 2-5.

There can be only one TITLE statement in each program division. You can place TITLE statements anywhere within NDL program divisions after the NFILE or LFILE statements.

Use of the TITLE statement is optional. If you do not use the TITLE statement, the file name specified on the NFILE or LFILE statement is used as the title of the NCF or LCF. A TITLE statement without a character string produces an untitled listing. Character strings must not exceed 45 characters.

Network Division

    NFILE statement ◄─── Repeat for each new
                                                                    network configuration
                                                                    file required.

    [TITLE statement]

        TRUNK statement ◄── Repeat for each line
                                                                   linking two NPUs.

        NPU statement

        SUPLINK statement ◄────────────── Repeat for each super-
                                          visory logical link.

            COUPLER statement   ┐ ◄─Repeat for each coupler.   ◄─Repeat for each NPU.

                LOGLINK statement ◄──────── Repeat for each logical
                                            link starting in this
                                          ┘ coupler.

            LINE statement   ┐
                                       │
            [GROUP statement]   ◄──────────────────────────── Repeat for each line.

                TERMINAL statement   ┐ ◄─ Repeat for each terminal
                                       │   connected to this line.

                    DEVICE statement ◄─┤ ── Repeat for each device
                                       ┘   connected to this terminal.

                [TERMDEV statement] ◄───── Repeat for each terminal
                                       ┘   connected to this line.

        NPU statement
            •
            •
            •

Local Division

    LFILE statement ◄─── Repeat for each new
                                                                    local configuration
                                                                    file required.

    [TITLE statement]

        [USER statement] ◄─────────────── Repeat for each terminal
                                          that requires an auto-
                                          matic or default login.

        [APPL statement] ◄─────────────── Repeat for each application
                                          program with special access
                                          requirements in the host.

        [OUTCALL statement] ◄──────────── Repeat for each path that
                                          can be used for requests
                                          for connection to another
                                          application program.

        [INCALL statement] ◄───────────── Repeat for each application
                                          program that can accept
                                          requests for connection from
                                          another application program.

    END statement ◄─── Only one per NDL
                                                                    program.

Figure 2-4. NDL Program Statement Hierarchy

```
TITLE[string].


string   Character string up to 45 characters
         long; includes any blanks following a
         comma separator. The string can
         contain any characters in the 6-bit
         display code set except for a period;
         a period terminates the title defini-
         tion. The default string is 45
         blanks.
```

Figure 2-5. TITLE Statement Format

An example of a valid TITLE statement is

   TITLE, SUNNYVALE CLOSED SHOP MHR NCF.

In this example, the listings and the network
configuration file produced by the NDL program
would contain the title line shown, identifying the
CDC Sunnyvale facility environment NCF used by the
CDC CYBER 170 closed-shop multiple-host network.

COMMENT STATEMENT

A COMMENT statement permits you to insert text
comments into NDL input. The NDL processor copies
these comments, without editing or interpretation,
into the output listing of your NDL source state-
ments. The format of the COMMENT statement is
shown in figure 2-6.

```
⎰COMMENT⎱ [,string] [.]
⎱ *    ⎰


string   Character string up to 89 characters
         long (asterisk form) or up to 82
         characters long (COMMENT form). The
         string can contain any characters in
         the 6-bit display code set. A
         comment string terminates at the end
         of a line, regardless of whether or
         not there is a terminating period.
         There is no default for string.

Comments that appear after the terminating
period of a statement must be in the form:

    *string
```

Figure 2-6. COMMENT Statement Format

Use of the COMMENT statement is optional. COMMENT
statements can be placed anywhere within NDL pro-
gram divisions.

An example of a valid COMMENT statement is

   COMMENT THIS IS A VALID COMMENT STATEMENT

DEFINE STATEMENT

A DEFINE statement allows you to assign a label
(identifier) to a set of keyword and value declara-
tions that is cumbersome to enter repeatedly. The

set of declarations is then stored as a character
string for insertion in subsequent parts of the NDL
program; the string is inserted by placing the
identifier in statements where the stored declara-
tion set is needed.

Use of the DEFINE statement is optional. The format
of the DEFINE statement is shown in figure 2-7.

```
defname:  DEFINE[string.]


defname     The identifier assigned to the
            character string. This is a
            name of one to seven letters or
            digits, the first of which must
            be a letter. The identifier
            should not be an NDL reserved
            word (see appendix D) and cannot
            be an NDL statement name or
            keyword. This name is required;
            there is no default value.

string      The character string for which
            the defname identifier is
            substituted when writing other
            NDL statements. This character
            string must contain complete
            value declarations and can
            contain commas, equals signs,
            blanks, and any letters or
            digits; it cannot contain a
            period, because a period is
            interpreted as the end of the
            DEFINE statement. The string
            can contain NDL reserved words
            but cannot contain NDL statement
            names or another DEFINE name.
```

Figure 2-7. DEFINE Statement Format

A DEFINE statement must precede any other statement
that uses the identifier specified in the DEFINE
statement. The defname identifier can only be used
in an NDL statement where the NDL processor expects
to find a keyword.

DEFINE statements can be placed anywhere within NDL
program divisions after the NFILE or LFILE state-
ments. DEFINE statements apply only to the division
in which they occur.

Nested DEFINE statements are not permitted. The
identifier of one DEFINE statement cannot be used
in the character string that defines the declaration
set of the same or another DEFINE statement. The
character string of a DEFINE statement cannot
contain another DEFINE statement.

As an example of this statement's function, consider
the following portion of an NDL program:

   LTYPE3: DEFINE LTYPE=A1,TIPTYPE=ASYNC.
   PORTA: DEFINE PORT=1.
      .
      .
      .
   LINE5: LINE PORTA, LTYPE3.

The first DEFINE statement permits the identifier
LTYPE3 to be substituted in subsequent NDL state-
ments (such as the last statement in this example)

for the character string LTYPE=A1, TIPTYPE=ASYNC. Note that the LINE statement is still terminated by a period because this character string is considered to end with the C, not with the period that terminates the DEFINE statement.

The second DEFINE statement equates the identifier PORTA to the character string PORT=1, so that the LINE statement is actually interpreted by the NDL processor as if it had been written

    LINE5: LINE PORT=1,LTYPE=A1,TIPTYPE=ASYNC.

DEFINE statements receive separate diagnostic treatment when the NDL program is processed. However, if a DEFINE statement contains an NDL coding error in its character string, the error might not be diagnosed until an attempt is made to expand the defname identifier.

The following combination of statements would not be recognized as expanding into a valid NDL statement, because the reserved statement name LINE cannot be found in the unexpanded version of LINE5:

    LINEA: DEFINE LINE,PORT=1.
    LTYPE3: DEFINE LTYPE=A2.
        .
        .
        .
    LINE5: LINEA,LTYPE3,TIPTYPE=ASYNC.

END STATEMENT

An END statement must explicitly terminate the last division in the NDL program. The NDL processor requires this statement to determine that the input file is complete.

This statement has the format shown in figure 2-8.

```
END.
```

Figure 2-8. END Statement Format

The network division describes the physical and logical configuration of the network. This includes a description of the processors within the network, the physical and logical links between processors, the communication lines and terminals, and the initial status of each of these elements.

Each host in the network running a copy of the Network Supervisor (NS) program or the Communications Supervisor (CS) program must also have a network configuration file (NCF). All of the NCFs for hosts belonging to the same site must be identical with one another except for line and terminal definitions. This means that the node information (couplers, logical links, and trunks) must all be the same for each NCF. The line and terminal definitions in the NCF used to load and configure an NPU must agree with the one used to supervise it.

Within a site, all NPUs must run the same release level of CCP. For sites running different release versions of host software and CCP, the level of NCF must correspond to the level of CCP being used. The level of NCF is defined as the release level of the NDLP that was used to create it. The level of NCF, however, does not need to match the level of NS or CS using it.

You can omit the network division from an NDL program if no network configuration file is to be created; otherwise, one network configuration file is created for each network division you define.

The network division does not apply to CDCNET networks.

NFILE STATEMENT

The NFILE statement identifies the network configuration file to be created. It must be the first statement, other than a COMMENT statement, in the network division.

The name of the NOS local file specified in the NFILE statement should not be the same as that of another local file assigned to your job. The network configuration file is described in more detail in section 9.

The format of the NFILE statement is shown in figure 3-1. An example of a valid NFILE statement is

    NETFILE: NFILE.

This statement creates a network configuration file with the NOS local file name NETFILE.

```
nfile:  NFILE.


nfile   The local file name of the new net-
        work configuration file to be
        created. This name is required and
        must conform to NOS file name con-
        ventions; there is no default value.
```

Figure 3-1. NFILE Statement Format

NETWORK NODE AND LINK DEFINITION STATEMENTS

Your NDL program's network division must describe all communication nodes, physical linkages, and logical linkages within the set of network processing units. You use the following statements to do this:

NPU statements define the loading, dumping, and network node requirements of each NPU.

SUPLINK statements define each logical link available as a path from an NPU to a Communications Supervisor for initiating, monitoring, and assisting NPU operations.

COUPLER statements define the physical linkages between each front-end NPU and a host computer data channel.

LOGLINK statements define the logical linkages between host nodes and network nodes within the set of NPUs or between one host node and another.

TRUNK statements define the physical linkages between NPUs.

Several of these statements require you to use the parameter values declared in other statements; that is, you must use the same value on two different statements. The NODE parameter also appears in several statements; however, values you define for NODE must be unique and should be assigned consecutively from 1.

NPU STATEMENT

The NPU statement names the NPU and defines the node number by which the NPU is to be referenced by the network software. There must be one NPU statement for every Network Processing Unit in the network. The format of the NPU statement is shown in figure 3-2.

The OPGO value used remains in effect for as long as the network definition is used. The DMP value used remains in effect until it is overridden by an operator DUMP command or LOAD command NDMP option. An operator LOAD command applies only to one loading operation on the NPU, so the DMP parameter value is again used when that operation is completed. An operator DUMP command remains in effect until the network is shut down, so the DMP parameter value is not used until the network is again initialized.

For example, consider the following NPU statement:

    NPUA: NPU, NODE=2, VARIANT=N3L, OPGO, DMP.

This definition describes NPUA (a front-end NPU with a node number of 2). NPUA is loaded according to the Network Supervisor program directives in the program initiation control block (PICB) called N3L within the network load file.

Because OPGO is used, an operator intervention requirement exists for NPUA whenever its loading process is completed. Before typing GO, the operator can enable or disable any of the network elements associated with the NPU. NPUA always establishes a supervisory relationship with a Communications Supervisor (CS) program and then requests operator permission (a GO command) before it begins to service its configured communication lines.

Because DMP is used, the long-term dump flag of the Network Supervisor (NS) for NPUA is set. The NS will dump the memory of NPUA each time NPUA is loaded, unless a NPU, a diagnostic or a host operator (NOP, DOP, or HOP) unsets the dump flag with a DMP command, or unless repeated loads occur within a short time span. (Repeated reloading during a short time span is presumed to result from repeated failures with the same cause, so multiple dumps are redundant.)

If DMP=NO were specified on the NPU definition statement, the long-term dump flag for NPUA would not be set at network initiation (unless the NOP or HOP set it with a command).

---

npu: NPU,NODE=node,VARIANT=picbname[,OPGO[=yn1],DMP[=yn2]].

npu	The element name of the NPU being described. This name is required; there is no default value.
node	The network node number (1 ≤ node ≤ 255) by which the network software references this NPU. All values declared for node must be unique within the current network definition and should be numbered consecutively. This number is required; there is no default value.
picbname	The name assigned to the program initiation control block (PICB) on the network load file for this NPU. The PICB contains directives to the Network Supervisor for dumping and loading a given NPU. This name consists of one through six alphabetic and numeric characters, the first of which must be alphabetic. The name is required; there is no default value. (Self-loading, self-configuring NPUs also must have a value declared for this parameter; use a dummy name.) See the NOS installation handbook for more information about this parameter.
yn1	An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter indicates whether an operator GO command is required to start NPU operation after loading is completed. If OPGO or OPGO=YES is specified, the HOP or controlling NOP must type GO each time the NPU is loaded before the NPU can become active. If OPGO is omitted or OPGO=NO is specified, operator intervention is not required to start the NPU. You should define an OPGO=YES value only if frequent configuration changes are anticipated for the NPU.
yn2	An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter indicates whether a long-term requirement exists for an NPU dump each time the NPU is loaded. If DMP or DMP=YES is specified, or if DMP is omitted, the NPU is dumped before it is loaded, unless the operation is overridden by a HOP or NOP command. If DMP=NO is specified, no dump occurs before the NPU is loaded, unless specifically requested by a HOP or NOP command. (Note that an NPU dump is essential for CDC to analyze NPU failure problems.)

Figure 3-2. NPU Statement Format

SUPLINK STATEMENT

The SUPLINK statement describes a supervision path between the NPU and a copy of CS which can supervise that NPU and defines the priority for use of that path. A supervision path, or supervisory link, corresponds to a logical link between the NPU and a host.

Only one supervisory link can be defined in each SUPLINK statement. If there is more than one logical link between an NPU and the host computer, each can appear in a separate SUPLINK statement; these statements must appear in descending order of priority. The first supervisory link defined has the highest priority, the second defined has the next highest priority, and so on. The supervisory links are used in the order of their priority if the logical link in a higher priority supervision path fails or the CS becomes available.

The SUPLINK statement must follow an NPU statement or another SUPLINK statement. The SUPLINK statement is optional if there is only one logical link between the NPU and a host. The format of the SUPLINK statement is shown in figure 3-3.

```
SUPLINK,LLNAME=loglink.


loglink      Element name of the logical link
             that is to be used as a super-
             visory link by the NPU. This
             name must appear on a LOGLINK
             statement within the same net-
             work division.
```

Figure 3-3. SUPLINK Statement Format

An example of the SUPLINK statement in an NDL program sequence is

```
NPUA: NPU, NODE=5, VARIANT=N3L, DMP=YES.
   SUPLINK, LLNAME=LGLNK1.
   CPLR1: COUPLER, NODE=1.
   LGLNK1: LOGLINK, NCNAME=NPUA.
```

This example defines the logical link called LGLNK1 as the only logical link for NPU supervisory message traffic with a Communications Supervisor program. LGLNK1 exists between coupler node 1 and network node 5 within the NPU called NPUA. For an NPU which has logical links to more than one host, you can define the logical link as a supervisory link only if the release level of the NCF being used on the host matches the release level of the CCP being run on the NPU.

COUPLER STATEMENT

The COUPLER statement identifies a channel coupler to an NPU. A front-end NPU can be equipped with one or two channel couplers connecting the NPU to a host. The format of the COUPLER statement is shown in figure 3-4.

One COUPLER statement is required for each coupler in the network. The COUPLER statement must follow the NPU statement defining the NPU that uses the coupler to function as a front-end NPU.

Each CDC host computer can support up to eight couplers on separate data channels. CDC host software identification of each coupler is determined by an entry in the NOS equipment status table (EST).

The value you use for the NODE declaration on each COUPLER statement must be the value assigned as the node number in the EST entry for the corresponding NPU coupler. The coupler's enabled or disabled status at network initiation is also controlled by this EST entry.

```
coupler:  COUPLER,NODE=node[,HNAME=hostnam,LOC=cplrloc].


coupler    The element name to assign to the front-end coupler being configured. This name is
           required; there is no default value.

node       The host node number (1 < node < 255) by which the network software references this
           coupler. This value must match the node number in the EST entry for the coupler. All
           values declared for NODE keywords must be unique within the current NDL program division
           and should be numbered consecutively. This number is required; there is no default value.

hostnam    The name assigned to the host to which the coupler connects. The hostnam value is 1 to 7
           letters or digits; the first character must be a letter. The hostnam you assign
           identifies the corresponding node number on the host availability display seen by the
           terminal user. This parameter is optional; if omitted, the default value is ASCII blanks.

cplrloc    The reserved word value (PRIMARY or SECOND) that indicates the coupler's location in the
           255x cabinet. PRIMARY specifies the coupler in the primary coupler slots and SECOND
           specifies the coupler in the secondary coupler slots. If an NPU has two couplers,
           LOC=SECOND is required for the secondary coupler. This parameter is optional; the default
           value is PRIMARY.
```

Figure 3-4. COUPLER Statement Format

The NODE value and the HNAME value appear as a set on the host availability display seen by the terminal user. The user can select connection to a host by specifying a node number or host name in a command to CCP; nonblank HNAME values are recommended to aid users in selecting the appropriate node.

The circuit boards for a coupler are located in slots marked as either primary or secondary within the upper card cage of the 255x series NPU cabinet. These markings are similar to port numbers for the couplers and are used in that manner for the LOC parameter.

If the NPU has one coupler, it can be located in either the primary or secondary slot. If the NPU has a coupler in the secondary slot, the coupler must be defined as a secondary coupler.

For example, suppose NPUA in SVLnet (figure 2-1 in section 2) contained two couplers, only one of which (the secondary one) belonged to the network shown. The circuit boards in the secondary slot would be cabled to host data channel number 4; the EST entry would associate node number 1 with data channel number 4. This coupler could be configured correctly only by the statement:

```
NPUA: NPU, NODE=5, VARIANT=N3L, DMP.
    CPLR1: COUPLER NODE=1, LOC=SECOND.
```

As used in the sample network configured by section 11, NPUA contains only one coupler. It therefore is configured correctly by the statement:

```
NPUA: NPU, NODE=5, VARIANT=N3L, DMP.
    CPLR1: COUPLER NODE=1,HNAME=SYS173.
```

Both examples define coupler CPLR1, which is physically contained in the cabinet of NPUA; NPUA uses CPLR1 to function as a front-end NPU. CPLR1 has a NOS EST entry of 1 for its host node number. Note that NPUA contains both a host node (node 1) for its coupler and a network node (node 5) for itself.

The COUPLER statement also defines a possible path the NPU can use for a request to be loaded and/or dumped by a host.

LOGLINK STATEMENT

The LOGLINK statements define a path over which terminal-to-application or application-to-application connections can be established. You must supply one LOGLINK statement for each logical link in your network. You need LOGLINK statements to define the logical link between a coupler and the NPU that contains the coupler and between a coupler and each NPU connected to the front-end NPU containing the coupler. You also need a LOGLINK statement to define each host-to-host logical link between two couplers if application programs in the two hosts are to communicate with each other.

Application programs in two hosts can communicate with each other without having a LOGLINK statement for the two couplers involved only if there is an intervening X.25 network between the two hosts. (Logical links cannot span X.25 public data networks, but always terminate at the NPU connected to the public data network.)

Only one logical link can be defined between any given pair of coupler and NPU or coupler node numbers. It is possible, however, to have more than one logical link between an NPU and a host computer. Figure 3-5 shows the logical links in SVLnet. Each logical link uses a separate trunk between the NPU and/or a different coupler.

Logical links can only be defined over paths that do not cross more than one trunk. In the example shown in figure 3-5, no logical link can exist between NPUC and coupler 11.

The LOGLINK statement must follow the COUPLER statement that defines the coupler for the host end of the logical link being defined. LOGLINK statements define which NPUs can access that coupler. For a coupler-to-coupler link, the LOGLINK statement can follow either of the COUPLER statements, but cannot appear after both. The format of the LOGLINK statement is shown in figure 3-6.

For example, in SVLnet (figure 2-1 in section 2), NPUA (node 5) contains coupler CPLR1 (node 1) and is connected to NPUE (node 9). NPUB (node 6) contains coupler CPLR2 (node 2) and is also connected to NPUE. NPUE has connections to several other front-end NPUs, giving its network node four possible logical links to the same host.

The host node in NPUA is node 1; the host node in NPUB is 2 (the host node number is always the number associated with the coupler). The following LOGLINK statements would properly configure two of the logical links involving NPUE as network node 9:

```
NPUA: NPU, NODE=5, VARIANT=N3L, DMP.
    CPLR1: COUPLER NODE=1,HNAME=SYS173.
        LINK1: LOGLINK NCNAME=NPUE.
    .
    .
    .
NPUB: NPU, NODE=6, VARIANT=N5L.
    CPLR2: COUPLER NODE=2,HNAME=SYS173.
        LINK2: LOGLINK NCNAME=NPUE.
    .
    .
    .
NPUE: NPU, NODE=9,...
```
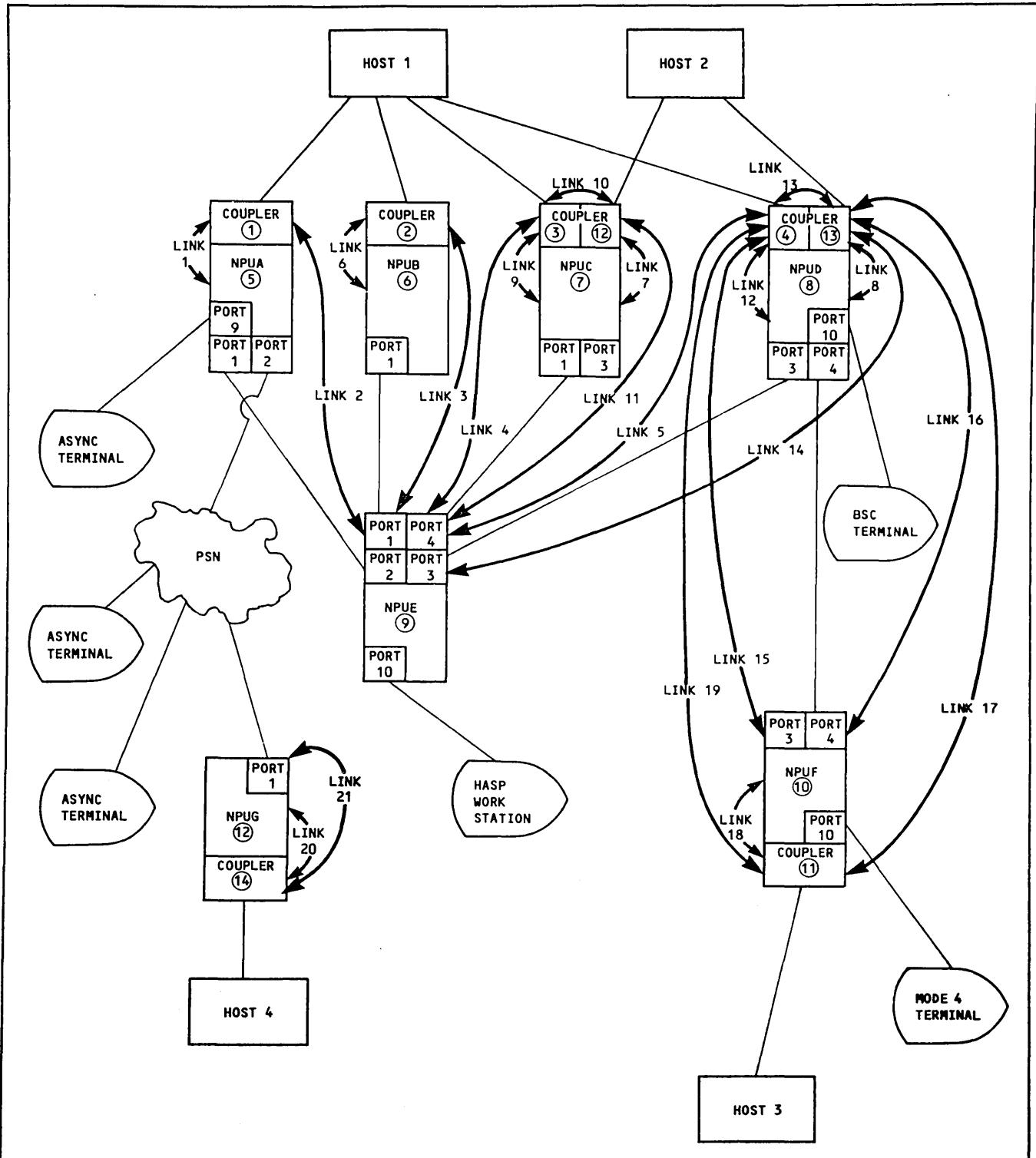
Figure 3-5. Logical Links

```
loglink:   LOGLINK,NCNAME=ncname[,DI[=yn]].


loglink      The element name to assign to the
             logical link you are
             configuring. This name is
             required; there is no default
             value.

ncname       The element name of the NPU or
             coupler that is the other end of
             this logical link. This name
             must be the element name used in
             an NPU or coupler statement in
             the same network division. This
             name is required; there is no
             default value.

yn           An optional reserved word value
             (YES or NO) for an optional
             stand-alone keyword; this
             parameter specifies the status
             assigned to this logical link at
             network initiation. When DI or
             DI=YES is specified, the logical
             link is disabled at network
             initiation and cannot be used
             until the HOP or NOP enables it.
             If DI is omitted or DI=NO is
             specified, the logical link is
             given an initial status of
             enabled unless the NOP or HOP
             specifies otherwise.
```

Figure 3-6. LOGLINK Statement Format

TRUNK STATEMENT

The TRUNK statement describes the trunk communi-
cation line that physically connects two NPUs.
There can be only one trunk defined between any
pair of NPUs in a network. Port numbers for all
trunks/lines should be assigned consecutively,
starting with one. Since the NPU reserves conti-
guous memory space for ports, this conserves memory
space.

The format of the TRUNK statement is shown in
figure 3-7. The recommended values for the FRAME
parameter are given in table 3-1.

Because the TRUNK statement contains all the
necessary information explicitly, it can be placed
anywhere within the NDL input stream after the NFILE
statement. We recommend, for ease of reading, that
you place all TRUNK statements at either the begin-
ning or the end of the division.

An example of a TRUNK statement is

    TRUNK1: TRUNK, N1=NPUA, N2=NPUE, P1=1, P2=2.
    NPUA: NPU, NODE=5, VARIANT=N3L, DMP.
        .
        .
        .
    NPUE: NPU, NODE=9, VARIANT=N5L.

TABLE 3-1. RECOMMENDED FRAME SIZES
FOR TRUNK LINES

Traffic is Mostly	Trunk Line Speed in Bits Per Second			
	4800 or Less	9600	19200	56000
Interactive	1050	1050	1050	1050
Batch and Interactive	256	500	1050	1050
Batch	1050	1050	1050	1050

The N1 and N2 values indicate that the trunk
connects the NPUs NPUA and NPUE. The P1 and P2
parameters indicate that the trunk is connected to
the Communications Line Adapter (CLA) port 1 in NPUA
and to port 2 in NPUE.

PROTOCOL-DEPENDENT DEFINITION STATEMENTS

The Network Definition Language allows you to
configure communication lines, terminals, and
devices using the following communication line
protocols:

    Asynchronous (IBM 2741- or teletypewriter-
    compatible devices)

    X.25 packet-switching network (teletypewriter-
    compatible devices)

    IBM HASP bisynchronous (HASP-compatible devices,
    including batch equipment)

    CDC Mode 4 synchronous (200-User-Terminal or
    Mode-4C-compatible devices, including batch
    equipment)

    IBM bisynchronous (IBM 2780-, 3780-, or
    3270-compatible devices, including batch
    equipment)

    Site-defined (any devices, compatible or incom-
    patible with any of the above)

Your NDL program's network division must describe
all communication lines, terminals, and devices
connected to the set of network processing units.
You use the following statements to do this:

    LINE or GROUP statements define the physical
    and logical service requirements of each com-
    munication line on each NPU.

    TERMINAL or TERMDEV statements define the
    clustering of devices connected to each com-
    munication line.

    DEVICE or TERMDEV statements define the physical
    and logical support requirements of each device
    within a terminal.

```
trunk:  TRUNK,N1=npu1,N2=npu2,P1=port1,P2=port2

     [,NOLOAD1[=yn1],NOLOAD2[=yn2],FRAME=frame,DI[=yn3]].
```

trunk   The element name to be assigned to the trunk communication line being defined. This name is required; there is no default value.

npu1   The element name for one of the two NPUs connected by the trunk. This name must appear on an NPU statement in the same network definition. This name is required; there is no default value.

npu2   The element name for the other of the two NPUs connected by the trunk. This name must appear on an NPU statement in the same network definition. This name is required; there is no default value.

port1   The hexadecimal number ($1 \leq$ port1 $\leq$ FE) of the port on npu1 to which this trunk is connected. This number cannot appear on another TRUNK or LINE statement (or within a GROUP statement expansion) for definitions applying to npu1 in the same network division; all trunk lines on an NPU must be assigned consecutively numbered ports starting at 1, and this number must be the thumbwheel number on the CLA for the trunk line. If npu1 is to be loaded over this trunk, $1 \leq$ port1 ≤ 4.

port2   The hexadecimal number ($1 \leq$ port2 $\leq$ FE) of the port on npu2 to which this trunk is connected. This number cannot appear on another TRUNK or LINE statement (or within a GROUP statement expansion) for definitions applying to npu2 in the same network division; all trunk lines on an NPU must be assigned consecutively numbered ports starting at 1, and this number must be the thumbwheel number on the CLA for the trunk line. If npu2 is to be loaded over this trunk, $1 \leq$ port2 ≤ 4.

yn1   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter prohibits loading npu1 over this trunk. If NOLOAD1 or NOLOAD1=YES is specified, npu1 cannot be loaded by npu2. If NOLOAD1 is omitted or NOLOAD1=NO is specified, npu1 will be loaded by npu2 whenever npu1 requests that operation. Usually a NO value is specified for trunks connecting NPUs if administrative responsibility for the two NPUs rests with a single site. If the two NPUs are administered by different sites, a YES value is specified.

yn2   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter prohibits loading npu2 over this trunk. If NOLOAD2 or NOLOAD2=YES is specified, npu2 cannot be loaded by npu1. If NOLOAD2 is omitted or NOLOAD2=NO is specified, npu2 will be loaded by npu1 whenever npu2 requests that operation.

frame   The approximate decimal number ($0 \leq$ frame ≤ 1200) of 8-bit character bytes that will be transmitted as one frame across this trunk. The value you should declare for this parameter depends on the type of data traffic across the trunk and the speed of data transmission on the line. Recommended values appear in table 3-1; declared values are rounded as follows:

Declared Frame Value	Value Used
0 thru 499	256
500 thru 1049	500
1050 thru 1200	1050

yn3   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter specifies the status assigned to this trunk at network initiation. When DI or DI=YES is specified, the trunk is disabled at network initiation and cannot be used until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the trunk is given an initial status of enabled unless the NOP or HOP specifies otherwise.

Figure 3-7. TRUNK Statement Format

These statements must contain value declarations to support hardware and protocols known to CDC but can also contain declarations to support site-defined hardware or protocols. Site-defined value declarations do not have the interdependencies of CDC-defined value declarations and receive less diagnostic inspection during NDL processor execution. You are responsible for proper agreement among values on these statements when one statement contains a site-defined value.

These five statements provide the network software with information that it transmits to CCP as a set of paired information bytes called field number/field value (FN/FV) pairs. Each of these statements (except the TERMINAL statement) can contain the keywords P90 through P99, identifying field numbers not used by the released version of CDC network software.

If your site has modified CCP software or has written its own Terminal Interface Program, these optional field number parameters allow you to specify the values to be transmitted for the corresponding field number when the terminal or device is configured. If you place the same field number parameter in more than one of the statements in a hierarchical set, each value specified is used separately. Each line and device has a separate set of field number/field value pairs associated with it.

For example, if you place P90 on the GROUP statement and on one of the DEVICE statements in the following set:

```
GROUP statement
    TERMINAL statement
        DEVICE statement
        DEVICE statement
        DEVICE statement
```

the value specified on the DEVICE statement is used as field 90 for the configuration transmitted to CCP for that device, and the value specified on the GROUP statement is used as field 90 for each communication line. No 90 value exists for the remaining devices.

The parameters required and the values allowed on the GROUP, LINE, TERMINAL, TERMDEV, and DEVICE statements depend upon the line protocol used. This section presents the general form of each statement. The following sections present the form of each statement and the parameters applicable to each line protocol:

Section 4 applies to asynchronous protocol lines.

Section 5 applies to X.25 protocol lines.

Section 6 applies to synchronous protocol lines.

Section 7 applies to site-defined protocol lines.

LINE STATEMENT

Each LINE statement defines one communication line between a terminal and the NPU. There must be one LINE statement for each CLA port on the NPU that supports terminal access. Figure 3-8 shows the general format of the LINE statement.

GROUP STATEMENT

You can use the GROUP statement in place of the LINE statement when you want to repeat a line definition a specified number of times. NDLP does not allow GROUP statements for X.25 protocol communication lines.

```
line:   LINE,PORT=port,LTYPE=ltype

    [,TIPTYPE=tiptype,AUTO[=yn1],]

    [XAUTO[=yn2],DI[=yn3],LSPEED=lspeed,]

    [AL=acclev,DFL=dfl,FRAME=frame,]

    [RTIME=timer,RCOUNT=count,]

    [NSVC=svcirc,PSN=psn,DCE[=yn4],]

    [DTEA=locadr,ARSPEED=yn5,]

    [IMDISC[=yn6],LCN=lcn,RC[=yn7],]

    [P90=fv90,...,P99=fv99].


Parameters are described in sections 4
through 7.
```

Figure 3-8. General LINE Statement Format

Using the GROUP statement can reduce the number of definitions you must provide in your NDL program without reducing the number of definitions provided in the network configuration file. The general format of the GROUP statement is shown in figure 3-9.

TERMINAL STATEMENT

The TERMINAL statement defines the characteristics of a multiple device workstation, such as a mode 4C terminal device cluster. This statement has the general format shown in figure 3-10.

DEVICE STATEMENT

The DEVICE statement defines the characteristics of a single terminal input and/or output device, such as a console or card reader. This statement has the general format shown in figure 3-11.

TERMDEV STATEMENT

The TERMDEV statement defines a terminal that consists of a single input and output device, such as an asynchronous console. This statement has the general format shown in figure 3-12.

```
group:  GROUP,PORT=port,LTYPE=ltype

        [,TIPTYPE=tiptype,AUTO[=yn1],]

        [XAUTO[=yn2],DI[=yn3],LSPEED=lspeed,]

        [AL=acclev,NI=iter,ARSPEED=yn4,]

        [IMDISC[=yn5],LCN=lcn,RC[=yn6],]

        [P90=fv90,...,P99=fv99].


Parameters are described in sections 4
through 7.
```

Figure 3-9. General GROUP Statement Format

```
TERMINAL[,STIP=stiptyp,TC=trmclas,]

         [CSET=charset,TSPEED=trmsped,]

         [CA=clsadr,RIC=yn1,]

         [CO=conord,BCF[=yn2],MREC=rec,]

         [W=pacwndw,CTYP=cirtyp,NCIR=numcir,]

         [PAD=string,EOF=yn3]

         [NEN=encir,COLLECT[=yn3]].


Parameters are described in sections 4
through 7.
```

Figure 3-10. General TERMINAL Statement Format

```
device:  DEVICE[,DT=devtyp,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

        [DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,DO=devord,STREAM=streamno,]

        [AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],LK=yn5,AB=ab,BR=br,CP=cp,BS=bs,B1=b1,]

        [B2=b2,CI=ci,CN=cn,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,EBO=ebo,]

        [ELX=elx,ELR=elr,ELO=elo,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,]

        [IN=in,LI=li,OP=op,OC=oc,PA=pa,PG=pg,PL=pl,PW=pw,RTS[=yn6],MCI=mc,MLI=ml,EOF=yn7,]

        [P90=fv90,...,P99=fv99].


Parameters are described in sections 4 through 7.
```

Figure 3-11. General DEVICE Statement Format

```
device:  TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,TSPEED=trmsped,CA=clsadr,RIC=yn1,]

        [CO=conord,BCF[=yn2],MREC=rec,W=pacwndw,NCIR=numcir,PAD=string,NEN=encir,]

        [COLLECT[=yn3],DT=devtyp,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

        [DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,DO=devord,STREAM=streamno,]

        [AUTOCON[yn4],PRI[=yn5],DI[=yn6],HN=node,HD[=yn7],LK=yn8,AB=ab,BR=br,CP=cp,BS=bs,B1=b1,]

        [B2=b2,CI=ci,CN=cn,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,EBO=ebo,]

        [ELX=elx,ELR=elr,ELO=elo,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,]

        [IN=in,LI=li,OP=op,OC=oc,PA=pa,PG=pg,PL=pl,PW=pw,RTS[=yn9],MCI=mc,MLI=ml,EOF=yn10,]

        [P90=fv90,...,P99=fv99].


Parameters are described in sections 4 through 7.
```

Figure 3-12. General TERMDEV Statement Format

The parameters required and the values allowed on the GROUP, LINE, TERMINAL, TERMDEV, and DEVICE statements depend upon the line protocol used. This section describes the form of a statement applicable to asynchronous protocol lines.

Asynchronous communication lines are those using the line types defined in table 4-1. Such lines are serviced by the CDC-supplied ASYNC TIP. If you use the site-defined line type, the ASYNC TIP is presumed to be modified by your site (if necessary) to support:

> A standard communication line adapter in a non-standard manner
>
> A specially built communication line adapter

LINE DEFINITION

You configure terminals on an asynchronous communication line by using the following statements:

> One LINE or GROUP statement that defines the line.
>
> One TERMINAL or TERMDEV statement for each potential terminal on the line (only one can access the line at a given time; multiple-drop asynchronous lines are not supported).
>
> One DEVICE statement for the device that accesses the line through each terminal; if you use a TERMDEV statement, you cannot also use a DEVICE statement.

LINE STATEMENT PARAMETERS

Each LINE statement defines one communication line between a terminal and the NPU. There must be one LINE statement for each CLA port on the NPU that supports terminal access.

Figure 4-1 shows the format of the LINE statement and the valid parameter values for CDC-defined asynchronous lines. Table 4-1 shows permitted line type values for the LTYPE parameter.

You use the AUTO or XAUTO parameter to define a communication line as having a fixed or an automatic recognition configuration. You can use either of these configuration types for a switchable (dialup) line or for a dedicated (hardwired) line.

If you define a fixed-configuration line by omitting AUTO and XAUTO or by specifying AUTO=NO or XAUTO=NO, you must know and specify certain characteristics of the terminal devices that will use that line. Only a terminal device with the specified characteristics can use that line.

The characteristics of a fixed-configuration line are:

> Line speed (LINE statement LSPEED parameter)
>
> SubTIPtype (TERMINAL or TERMDEV statement STIP parameter)
>
> Code and character set (TERMINAL or TERMDEV statement CSET parameter)
>
> Device type (TERMDEV or DEVICE statement DT parameter)

TABLE 4-1. ASYNCHRONOUS LINE TYPE DEFINITIONS

LTYPE Value	Transmission Mode	Transmission Operation	Circuit Type	Modem Type†	CLA Type	Maximum Speed, Bits per Second	Carrier Type
A1	Asynchronous	Full-duplex	Switched (Dialup)	RS232C, Bell 103E/113/ 212A compatible	2561-1	9600	Constant
A2	Asynchronous	Full-duplex	Dedicated (Hardwired)	RS232C, Bell 103E/113/ 212A compatible	2561-1	9600	Constant
A6 (For site-defined use)	Asynchronous	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned

†Modem types supported by each type of CLA may differ.

```
line:   LINE,PORT=port,LTYPE=ltype,TIPTYPE=tiptype[,AUTO[=yn1],XAUTO[=yn2],ARSPEED=arspeed,DI[=yn3],]

        [IMDISC[=yn4],LSPEED=lspeed,AL=acclev,RC[=yn5],P90=fv90,...,P99=fv99].
```

line
: The element name to be assigned to the communication line being defined. This name is required; there is no default value.

port
: The hexadecimal number (1 ≤ port ≤ FE) of the port to which this communication line connects on the NPU currently being defined. Within an NDL program, the port number is independent of the number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE (254 decimal). However, we strongly recommend that you assign port numbers consecutively, starting with 1. All values declared for PORT parameters must be unique within the current network definition of each NPU. The value specified for port cannot be the same as the number declared for a port in any other LINE or TRUNK statement (or within a GROUP statement expansion) for this NPU, and cannot be lower than the highest port number used by a trunk. This number is required; there is no default value.

ltype
: A reserved word value that identifies the type of communication line adapter/modem/circuit combination that is used on this line. This word must be supplied; there is no default value. The legal words for this value declaration are described in table 4-1.

tiptype
: A reserved word value that identifies the type of CCP Terminal Interface Program required for the terminals on this line. This word is required for lines being configured to service an asynchronous protocol terminal. The legal value for this declaration is:

    ASYNC   Asynchronous protocol TIP is required.

yn1
: An optional reserved word value (YES or NO) for an optional stand-alone keyword. This parameter specifies whether this communication line is configured for automatic recognition of low line speeds (110 thru 2400 bits per second) and/or recognition of protocol subTIPtype by CCP whenever a terminal is connected to the line. When AUTO or AUTO=YES is specified, the line is configured for automatic recognition. When automatic recognition is performed, CCP attempts to determine as many definition parameters as possible at the time the line becomes active; recognized parameters are compared against parameters specified in the network definition file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or if AUTO=NO or XAUTO=NO is specified, CCP is not allowed to perform automatic recognition for the line; the line has a fixed configuration, and each TERMINAL and DEVICE or TERMDEV statement associated with this line must explicitly declare all addressing and protocol parameters legal for that terminal.

yn2
: An optional reserved word value (YES or NO) for an optional stand-alone keyword. This parameter specifies whether this communication line is configured for automatic recognition of high line speeds (600 thru 9600 bits per second) and/or recognition of protocol subTIPtype by CCP whenever a terminal is connected to the line. When XAUTO or XAUTO=YES is specified, the line is configured for automatic recognition. When automatic recognition is performed, CCP attempts to determine as many definition parameters as possible at the time the line becomes active; recognized parameters are compared against parameters specified in the network definition file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or if AUTO=NO or XAUTO=NO is specified, CCP is not allowed to perform automatic recognition for the line; the line has a fixed configuration, and each TERMINAL and DEVICE or TERMDEV statement associated with this line must explicitly declare all addressing and protocol parameters legal for that terminal.

arspeed
: An optional reserved word value (YES or NO) indicating whether the user can change the line speed by using the AR TIP command. The default value of YES is used if AUTO or XAUTO is specified. The default value of NO is used if AUTO or XAUTO is not specified.

yn3
: An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter specifies the status assigned to this communication line at network initiation. When DI or DI=YES is specified, the line is disabled at network initiation and cannot be used until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is given an initial status of enabled unless the NOP or HOP specifies otherwise.

yn4
: An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on the line is disconnected immediately after the terminal has logged out (disconnected) from the host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is disconnected only after a two-minute timer expires.

Figure 4-1. Asynchronous LINE Statement Format (Sheet 1 of 2)

lspeed	The baud rate used by the modems or devices accessing this line. When the line is configured for automatic recognition (AUTO or XAUTO is declared), this parameter cannot be used. This parameter is optional when the line is not configured for automatic recognition of terminals. The following values are recognized:

110	300 default	2400	19200
134	600	4800	38400
150	1200	9600	

This value can be changed by the user if ARSPEED=YES. When lspeed is not declared for a line configured without automatic recognition, the default of 300 baud is used.

acclev	The decimal access level limit ($0 \le$ acclev ≤ 7) for the line. The lowest access level is 0 (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level limit is associated with the line (equivalent to specifying AL=0).

yn5	An optional reserved word value (YES or NO) which specifies whether the reconfiguration indicator is enabled for this line. When RC or RC=YES is specified, the terminal characteristics are reset to their original NDL values (or to a default if no NDL values were specified) when the terminal disconnects from a host. When RC is omitted or RC=NO is specified, the reconfiguration indicator is not enabled for this line.

fv9i	The hexadecimal field value ($0 \le$ fv9i $\le$ FF) to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this line. The released version of the CDC-written ASYNC TIP does not use any of these ten field number/field value pairs.

Figure 4-1. Asynchronous LINE Statement Format (Sheet 2 of 2)

If you define an automatic recognition line by specifying AUTO, XAUTO, XAUTO=YES, or AUTO=YES, CCP determines these characteristics when a terminal becomes active on the line. This allows more flexible access to the network.

A terminal on an automatic recognition line is not completely configured until it becomes active. At that time, the Communications Supervisor (CS) compares the determined characteristics against the characteristics you have defined for each terminal configured on the line. CS uses the first terminal definition that matches to finish configuring the terminal. CCP then services the terminal device according to the finished configuration.

Only a terminal that completely matches all declared values can use the line. It must operate with the characteristics you declare for any required or optional parameters in your terminal definition, or the terminal user must change its characteristics to match. If you declare AUTOREC instead of values for all of the automatically recognized parameters, then any terminal will match the terminal definition.

If you declare values instead of AUTOREC for some of the automatically recognized parameters, then any terminal that successfully accesses the line and has characteristics matching the declared values will match your terminal definition. If you declare values instead of AUTOREC for all of the automatically recognized parameters, then only a terminal with automatically recognized characteristics that are the same as your values will match the terminal definition.

You can specify more than one terminal definition on an automatic recognition line. Each definition should vary from the others in one of the automatically recognized characteristics; unless differences exist, CS will not use any definition other than the first one.

Using automatic recognition increases the number of terminal devices that potentially can use the line. You can configure a switchable line for automatic recognition with more logical terminal devices than the physical terminal devices that simultaneously can access it; only one physical device can access the line at a time.

For example, the statement

    LN1A: LINE PORT=9,LTYPE=A1,TIPTYPE=ASYNC,AUTO.

defines a switchable (dialup) communication line (on NPUA in SVLnet, figure 2-1 of section 2) for automatic recognition of any asynchronous protocol terminal with line speed less than or equal to 2400 b/s. The line is identified as LN1A, connects to the NPU at port 9, and is enabled by default at network initiation.

If the fixed-configuration form of the statement

    LN1A: LINE PORT=9,LTYPE=A1,TIPTYPE=ASYNC.

were used instead, all of the terminals capable of accessing this switchable line would have to use the explicit configuration provided by the same TERMINAL and DEVICE statement set.

The following two LINE statements are equivalent:

    LN1A: LINE PORT=9,LTYPE=A1,TIPTYPE=ASYNC,
       LSPEED=300.

    LN1A: LINE PORT=9,LTYPE=A1,TIPTYPE=ASYNC.

Both statements define a switchable, enabled, fixed-configuration asynchronous line accessing the NPU (NPUA in SVLnet) through port 9 and using modems at 300 baud (the default rate). The terminal actually accessing this line must use the same explicit configuration as any other that can potentially access it because the line is not configured for automatic recognition.

The network configuration defined for systems containing Centronics 533/536 printers must include definitions for these printers. There may be up to twelve printers connected to PSU, consisting of any mixture of 533/536 and Hitachi 585 printers. However, if there are more than eight 533/536 printers or more than four 585 printers, changes to the PSU EVFU file are required. Network configuration directives in the Network Division are required only for 533/536 printers, while USER statements in the Local Division are required for all printers connected to PSU.

The following statements may be used for the Network Division definitions:

```
LCENTR:    DEFINE LTYPE=A2,TIPTYPE=ASYNC,
           LSPEED=9600.
TCENTR:    DEFINE TC-721,AUTOCON,LK=YES,OC=YES,
           PA=0,PL=0,PW=0.

linex:   LINE LCENTR,PORT=portx.
devicex: TERMDEV TCENTR,HN=hostnode.
```

where linex is the line name for the printer, portx is the port number for the printer line, devicex is the device name for the printer, and hostnode is

the NODE number of the host coupler to which the printer is connected.

GROUP STATEMENT PARAMETERS

You can use the GROUP statement in place of the LINE statement when you want to repeat a line definition a specified number of times.

NDLP creates identical line, terminal, and device definitions (except for port number, line name, and device name) the number of times specified by the NI parameter. NDLP generates unique port numbers by incrementing the previously defined port number by one. NDLP generates unique line and device element names by adding the LINE's port number in two-digit hexadecimal form to the one- to five-character root element name you supply.

Figure 4-2 shows the format of the GROUP statement and the valid parameter values for CDC-defined asynchronous lines. Table 4-1 shows permitted line type values for the LTYPE parameter.

Using the GROUP statement can reduce the number of definitions you must provide in your NDL program

---

group: GROUP,PORT=port,LTYPE=ltype,TIPTYPE=tiptype[,AUTO[=yn1],XAUTO[=yn2],ARSPEED=arspeed,DI[=yn3],]

[IMDISC[=yn4],LSPEED=lspeed,AL=acclev,RC[=yn5],P90=fv90,...,P99=fv99,NI=iter].

group	The root name to use for generating the element names assigned to the communication lines and devices being defined. This name, which cannot be longer than five characters, must result in element names that are unique within the network division currently being described. This name is required; there is no default value.
port	The hexadecimal number (1 ≤ port ≤ FE) of the port to which the first of these communication lines connects on the NPU currently being defined. Within an NDL program, the port number is independent of the number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE (254 decimal). However, we strongly recommend that you assign port numbers consecutively, starting with 1. All values declared for PORT parameters must be unique within the current network definition of each NPU. The value specified for port cannot be the same as the number declared for a port in any LINE or TRUNK statement (or within any other GROUP statement expansion) for this NPU, and cannot be lower than the highest port number used by a trunk. This number is required; there is no default value.
ltype	A reserved word value that identifies the type of communication line adapter/modem/circuit combination that is used on this line. This word must be supplied; there is no default value. The legal words for this value declaration are described in table 4-1.
tiptype	A reserved word value that identifies the type of Terminal Interface Program required for the terminals on this line. This word is required; there is no default value. The valid value for this declaration is:
	ASYNC Asynchronous protocol TIP is required.
yn1	An optional reserved word value (YES or NO) for an optional stand-alone keyword. This parameter specifies whether this communication line is configured for automatic recognition of low line speeds (110 through 2400 bits per second) and/or recognition of protocol subTIPtype by CCP whenever a terminal is connected to the line. When AUTO or AUTO=YES is specified, the line is configured for automatic recognition. When automatic recognition is performed, CCP attempts to determine as many definition parameters as possible at the time the line becomes active; recognized parameters are compared against parameters specified in the network definition file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or if AUTO=NO or XAUTO=NO is specified, CCP is not allowed to perform automatic recognition for the line; the line has a fixed configuration, and each TERMINAL, DEVICE, or TERMDEV statement associated with this line must explicitly declare all addressing and protocol parameters legal for that terminal.

Figure 4-2. Asynchronous GROUP Statement Format (Sheet 1 of 2)

yn2	An optional reserved word value (YES or NO) for an optional stand-alone keyword. This parameter specifies whether this communication line is configured for automatic recognition of high line speeds (600 thru 9600 bits per second) and/or recognition of protocol subTIPtype by CCP whenever a terminal is connected to the line. When XAUTO or XAUTO=YES is specified, the line is configured for automatic recognition. When automatic recognition is performed, CCP attempts to determine as many definition parameters as possible at the time the line becomes active; recognized parameters are compared against parameters specified in the network definition file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or if AUTO=NO or XAUTO=NO is specified, CCP is not allowed to perform automatic recognition for the line; the line has a fixed configuration, and each TERMINAL and DEVICE or TERMDEV statement associated with this line must explicitly declare all addressing and protocol parameters legal for that terminal.
arspeed	An optional reserved word value (YES or NO) indicating whether the user can change the line speed by using the AR TIP command. The default value of YES is used if AUTO or XAUTO is specified. The default value of NO is used if AUTO or XAUTO is not specified.
yn3	An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter specifies the status assigned to this communication line at network initiation. When DI or DI=YES is specified, the line is disabled at network initiation and cannot be used until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is given an initial status of enabled unless the NOP or HOP specifies otherwise.
yn4	An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on the line is disconnected immediately after the terminal has logged out (disconnected) from the host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is disconnected only after a two-minute timer expires.
lspeed	The baud rate used by the modems or devices accessing this line. When the line is configured for automatic recognition (AUTO or XAUTO is declared), this parameter cannot be used. This parameter is optional when the line is not configured for automatic recognition of terminals; the following values are recognized:

110   300 default   2400   19200
134   600   4800   38400
150   1200   9600

This value can be changed by the user if ARSPEED=YES. When lspeed is not declared for a line configured without automatic recognition, the default of 300 baud is used.

acclev	The decimal access level limit (0 ≤ acclev ≤ 7) for the line. The lowest access level is 0 (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level limit is associated with the line (equivalent to specifying AL=0).
yn5	An optional reserved word value (YES or NO) which specifies whether the reconfiguration indicator is enabled for this line. When RC or RC=YES is specified, the terminal characteristics are reset to their original NDL values (or to a default if no NDL values were specified) when the terminal disconnects from a host. When RC is omitted or RC=NO is specified, the reconfiguration indicator is not enabled for this line.
fv9i	The hexadecimal field value (0 ≤ fv9i ≤ FF) to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this line. The released version of the CDC-written ASYNC TIP does not use any of these ten field number/field value pairs.
iter	The decimal number (1 ≤ iter ≤ 254) of iterations of this line definition and of all TERMINAL, TERMDEV, or DEVICE statements following this GROUP statement. This parameter is optional; the default is 1.

Figure 4-2. Asynchronous GROUP Statement Format (Sheet 2 of 2)

without reducing the number of definitions provided in the network configuration file. An example of GROUP statement use is shown in figure 4-3.

You use the AUTO or XAUTO parameter to define a communication line as having a fixed or an automatic recognition configuration. You can use either of these configuration types for a dialup line or for a dedicated (hardwired) line.

If you define a fixed-configuration line by omitting AUTO and XAUTO or by specifying AUTO=NO or XAUTO=NO, you must know and specify certain characteristics of the terminal device that will use the line. Only a terminal device with the specified characteristics can use that line.

The characteristics of a fixed-configuration line are:

Line speed (GROUP statement LSPEED parameter)

```
The following set of statements:

    LN1A: GROUP,PORT=9,LTYPE=A1,
        TIPTYPE=ASYNC,LSPEED=300,NI=5.
            DEVA:TERMDEV,TC=713,PRI.


is interpreted by the NDL processor as if it had
been written:

    LN1A09: LINE,PORT=9,LTYPE=A1,
        TIPTYPE=ASYNC,LSPEED=300.
            DEVA09: TERMDEV,TC=713,PRI=YES.

    LN1A0A: LINE,PORT=0A,LTYPE=A1,
        TIPTYPE=ASYNC,LSPEED=300.
            DEVA0A: TERMDEV,TC=713,PRI=YES.

    LN1A0B: LINE,PORT=0B,LTYPE=A1,
        TIPTYPE=ASYNC,LSPEED=300.
            DEVA0B: TERMDEV,TC=713,PRI=YES.

    LN1A0C: LINE,PORT=0C,LTYPE=A1,
        TIPTYPE=ASYNC,LSPEED=300.
            DEVA0C: TERMDEV,TC=713,PRI=YES.

    LN1A0D: LINE,PORT=0D,LTYPE=A1,
        TIPTYPE=ASYNC,LSPEED=300.
            DEVA0D: TERMDEV,TC=713,PRI=YES.
```

Figure 4-3.   Asynchronous GROUP Statement
Expansion

SubTIPtype (TERMINAL or TERMDEV statement STIP
parameter)

Code and character set (TERMINAL or TERMDEV
statement CSET parameter)

Device type (TERMDEV or DEVICE statement DT
parameter)

If you define an automatic recognition line by
specifying AUTO, XAUTO, XAUTO=YES, or AUTO=YES, CCP
determines these characteristics when a terminal
becomes active on the line. This allows more
flexible access to the network.

A terminal on an automatic recognition line is not
completely configured until it becomes active. At
that time, the Communications Supervisor (CS) com-
pares the determined characteristics against the
characteristics you have defined for each terminal
configured on the line. CS uses the first terminal
definition that matches to finish configuring the
terminal. CCP then services the terminal devices
according to the finished configuration.

Only a terminal that completely matches all declared
values can use the line. It must operate with the
characteristics you declare for any required or
optional parameters in your terminal definition, or
the terminal user must change those characteristics
to match. If you declare AUTOREC instead of values
for all of the automatically recognized parameters,
then any terminal with the automatically recognized
characteristics will match the terminal definition.

If you declare values instead of AUTOREC for some
of the automatically recognized parameters, then
any terminal that successfully accesses the line
and has characteristics matching the declared
values will match your terminal definition. If you
declare values ·instead of AUTOREC for all of the
automatically recognized parameters, then only a
terminal with automatically recognized characteris-
tics that are the same as your values will match
the terminal definition.

You can specify more than one terminal definition
on an automatic recognition line. Each definition
should vary from the others in one of the auto-
matically recognized characteristics; unless dif-
ferences exist, CS will not use any definition other
than the first one.

Using automatic recognition increases the number of
terminal devices that potentially can use the line.
You can configure a switchable line for automatic
recognition with more logical terminal devices than
the physical terminal devices that simultaneously
can access it; only one physical device can access
the line at a time.

For example, the statement

    LN1A: GROUP,PORT=9,LTYPE=A1,TIPTYPE=ASYNC,
        AUTO,NI=2.

defines two switchable (dialup) communication lines
(not shown in figure 2-1 of section 2) for automatic
recognition of asynchronous protocol terminals.
The lines are identified as LN1A09 and LN1A0A, con-
nect to the NPU at ports 9 and 0A, and are enabled
by default at network initiation.

If the fixed-configuration form of the statement

    LN1A: GROUP,PORT=9,LTYPE=A1,TIPTYPE=ASYNC,NI=2.

were used instead, all of the terminals capable of
accessing these switchable lines would have to use
the same explicit configuration provided by the
TERMINAL and DEVICE statement set following the
GROUP statement.

The following two GROUP statements are equivalent:

    LN1A: GROUP,PORT=9,LTYPE=A1,TIPTYPE=ASYNC,
        LSPEED=300,NI=5.

    LN1A: GROUP,PORT=9,LTYPE=A1,TIPTYPE=ASYNC,
        NI=5.

Each statement defines five switchable, enabled,
fixed-configuration asynchronous lines accessing
the NPU at ports 9 through 0D and using modems at
300 baud (the default rate). The terminals actually
accessing these lines must use the same explicit
configuration as any that can potentially access
the lines, because the lines are not configured for
automatic recognition.

TERMINAL DEFINITIONS

On an asynchronous communication line, a terminal
definition is the first portion of a device defini-
tion. An asynchronous terminal consists of only
one device, normally a console; terminal and device
are synonymous.

If you defined the communication line as a fixed-configuration line, you can declare only one TERMINAL or one TERMDEV statement for the line. If you defined the line as an automatic recognition line, you can declare up to 255 TERMINAL statements or TERMDEV statements for the line (only one can be connected at a given time).

Figures 4-4 and 4-5 present the formats of these statements for terminals on CDC-defined asynchronous lines. The following TERMINAL and TERMDEV parameters provide the terminal definition for the line:

    CSET
    RIC
    STIP
    TC
    TSPEED

```
TERMINAL[,STIP=stiptyp,TC=trmclas,]

    [CSET=charset,TSPEED=trmsped,]

    [RIC=yn1].


Parameters are described in the text.
```

Figure 4-4. TERMINAL Statement Format
for Communication Lines of TIPTYPE=ASYNC

The STIP and TC parameters are described first because they are related and affect the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

STIP PARAMETER

The STIP parameter specifies the subTIPtype required to support the terminal definition. This parameter is optional if the TC parameter specifies a CDC-defined terminal class. If STIP is omitted, the TC parameter must be specified and cannot be CCP or TC28 through TC31.

When you declare a value for STIP, it must be one of the following reserved words:

AUTOREC   Specifies that the subTIPtype appropriate for the terminal class should be used (valid only if TC is specified); using this value is equivalent to omitting the parameter.

N2741   Identifies an asynchronous terminal that does not use IBM 2741 protocol.

2741   Identifies an asynchronous terminal that uses IBM 2741 protocol.

If you omit this parameter or use the value AUTOREC, the default is selected according to the terminal class declared. The defaults are:

TC Value	STIP Value Used
M33	N2741
713	N2741
721	N2741
M40	N2741
H2000	N2741
X364	N2741
T4014	N2741
2741	2741

```
device:  TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,TSPEED=trmsped,RIC=yn1,DT=devtyp,]

    [ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [AUTOCON[=yn2],PRI[=yn3],DI[=yn4],HN=node,HD[=yn5],LK=yn6,AB=ab,BR=br,BS=bs,B1=b1,B2=b2,]

    [CI=ci,CN=cn,CP=cp,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,]

    [ELX=elx,ELR=elr,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,]

    [IN=in,LI=li,OP=op,OC=oc,PA=pa,PG=pg,PL=pl,PW=pw,RTS[=yn7],MCI=mc,MLI=ml,]

    [P90=fv90,... ,P99=fv99].


device    The element name of the terminal device being defined. This name can be one through seven
          characters long. The first character must be a letter; the other characters can be
          letters or digits. If this statement is used within the set following a GROUP statement,
          the device name cannot be longer than five characters. This is the name used by the host
          or NPU operator to monitor and control the device. This name is required; there is no
          default value.

All other parameters are described in the text.
```

Figure 4-5. TERMDEV Statement Format for Communication Lines of TIPTYPE=ASYNC

TC PARAMETER

The TC parameter specifies the terminal class appropriate for the device. Supported devices are grouped into terminal classes, according to their hardware characteristics.

Each CDC-defined terminal class has the default characteristics associated with an archetype terminal. Each CDC-defined terminal class has a range of possible characteristics. These ranges determine the values legal for parameters you specify on the TERMDEV or DEVICE statements.

Because the network cannot recognize differences among some terminal classes, a default terminal class exists for each subcategory of the TIP type (the subTIPtype, defined by the STIP parameter of the TERMINAL or TERMDEV statement); these defaults can be used when you declare a STIP parameter.

The mnemonics identifying the terminal classes are derived from the identification of the archetype terminals defining the classes, with the following exceptions: you can use TC28, TC29, TC30, and TC31 to identify site-defined terminal classes with no default characteristics, corresponding to reserved Network Access Method (NAM) terminal class numbers.

Because the characteristics of terminals in site-defined classes are unknown, TERMINAL, TERMDEV, or DEVICE statement parameter value ranges are checked against those valid for the STIP value specified. Any value declared for a terminal or device parameter is valid if the parameter and value conform to requirements of that subTIPtype.

This parameter is optional. When you declare a value for TC, it must be one of the following reserved words:

CCP Specifies that CCP is to provide the default terminal class appropriate for the subTIPtype; using this value is equivalent to omitting the parameter.

H2000 Identifies a terminal device compatible with Hazeltine 2000 series equipment emulating a teletypewriter.

M33 Identifies a terminal device compatible with Teletype Model 30 series equipment.

M40 Identifies a terminal device compatible with Teletype Model 40-2 equipment.

TC28 Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.

TC29 Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.

TC30 Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.

TC31 Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

T4014 Identifies a terminal device compatible with Tektronix 4014 series equipment emulating a teletypewriter.

X364 Identifies a terminal device compatible with the ANSI X3.64 standard (DEC VT100 or CDC 722-30).

2741 Identifies a terminal device compatible with IBM 2741 equipment.

713 Identifies a terminal device compatible with CDC Model 713, 722-10, 751, 752, or 756 series equipment.

721 Identifies a terminal device compatible with CDC Model 721 equipment.

If the TC parameter is omitted, the STIP parameter must be specified. If the TC parameter is specified with a value of CCP or TC28 through TC31, the STIP parameter must be specified. If the STIP parameter is omitted, the TC parameter must be specified and cannot have the values CCP or TC28 through TC31.

If the TC parameter is specified with a value of TC28, TC29, TC30, or TC31, then CCP must also be modified to accept these values. If the appropriate support code for TC28, TC29, TC30, or TC31 is not added to CCP, these values will be considered invalid by CCP.

If you omit the TC parameter or specify the value CCP, the default is selected according to the STIP value declared. The defaults are:

STIP Value	TC Value Used
N2741	M33
2741	2741

CSET PARAMETER

The CSET parameter specifies the code and character set of the terminal. This parameter is optional.

When you declare a value for CSET, it must be one of the following reserved words. For terminals in class 2741 or in site-defined terminal classes with a STIP value of 2741, the value can be:

AUTOREC The code set appropriate for the terminal class or determined by automatic recognition should be used; using this value is equivalent to omitting the parameter.

CORAPL IBM Correspondence code set with APL character set.

CORRES IBM Correspondence code with Correspondence code character set.

CSET15 Site-defined code and character set, identified within the network software as character set number 15.

EBCD IBM Extended BCD code and character set.

EBCDAPL IBM Extended BCD code set with APL character set.

For terminals in classes M33, 713, 721, X364, M40, H2000, and T4014, or in site-defined terminal classes with a STIP value of N2741, the value can be:

APLBP   ASCII code set with bit-pairing APL
         character set.

APLTP   ASCII code set with typewriter-pairing
         character set.

ASCII   ASCII code and character set.

AUTOREC The default code set for the terminal
         class or that is determined during
         automatic recognition of the terminal
         device should be used.

CSET15   Site-defined code and character set,
         identified within the network software
         as character set number 15.

If you specify CSET for a device on an automatic
recognition line, the network software performs an
additional match while identifying the terminal in
the configuration file. If you omit CSET, no
additional match is performed for a device on an
automatic recognition line, and the recognized
value is used.

For fixed-configuration lines, the default is
selected according to the STIP value declared or
used. The defaults are:

STIP Value	CSET Value Used
N2741	ASCII
2741	CORRES

The terminal user can change any value that you
specify.

RIC PARAMETER

The RIC parameter indicates whether the terminal
has restricted interactive capabilities. This
parameter is optional.

A terminal can have restricted interactive capa-
bilities for many reasons; for example, perhaps no
console device actually exists and interactive
dialog must occur through a site-defined device.

The effect this parameter has depends on the
application program the terminal uses. For example,
RBF does not output the READY prompt and does not
require a GO command to enable any defined passive
devices when RIC is specified.

Possible values for this parameter are:

NO   Indicates that the terminal has full
        interactive capabilities.

YES   Indicates that the terminal has limited
        interactive capabilities.

If you omit the RIC parameter, the default value of
NO is used.

TSPEED PARAMETER

The TSPEED parameter specifies the baud rate of the
terminal. This parameter is optional and valid
only when the communication line is defined for
automatic recognition.

If you specify a TSPEED value, the network software
performs an additional match while identifying the
terminal in the configuration file. Any terminal
accessing the communication line using the speed
you specify might match the terminal definition.

If AUTO is specified, the TSPEED parameter can have
the following values:

    110   134   150   300   600   1200   2400
    AUTOREC

If XAUTO is specified, the TSPEED parameter can
have the following values:

    600   1200   2400   4800   9600   AUTOREC

If you omit this parameter or use the value AUTOREC,
CCP determines the baud rate when it performs auto-
matic recognition for the terminal; CCP recognizes
any transmission speed valid for the communication
line.

DEVICE DEFINITIONS

You must provide one device definition for the
device that can access the communication line
through the terminal. If a TERMDEV statement is
used in place of the TERMINAL statement, a DEVICE
statement cannot be used. Each TERMINAL statement
can have only one DEVICE statement.

The released version of the network software does
not provide support for passive devices on asyn-
chronous lines. You can configure your own passive
devices using the terminal classes TC28 through
TC31, or you can connect them as secondary devices
to merely echo the input and output of the console
device.

Figures 4-5 and 4-6 present the formats of the
TERMDEV and DEVICE statements for devices on asyn-
chronous lines. The following DEVICE and TERMDEV
parameters provide the definition for an asynchro-
nous terminal device:

AB	DBZ	HN	PRI
ABL	DI	IC	PW
AUTOCON	DLC	IN	P90 through P99
BR	DLTO	LI	RTS
BS	DLX	LK	UBL
B1	DT	MCI	UBZ
B2	EBR	MLI	XBZ
CI	EBX	OC	XLC
CN	ELR	OP	XLTO
CP	ELX	PA	XLX
CT	EP	PG	XLY
DBL	HD	PL	

The DT parameter is described first because it
affects the validity, ranges, and defaults of the
other parameters. The other parameters are
described in alphabetical order.

```
device:  DEVICE[,DT=devtyp,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

    [XBZ=xmitsiz,AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],LK=yn5,AB=ab,BR=br,]

    [CP=cp,BS=bs,B1=b1,B2=b2,CI=ci,CN=cn,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,]

    [EBR=ebr,ELX=elx,ELR=elr,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,]

    [XLY=xly,IC=ic,IN=in,LI=li,OP=op,OC=oc,PA=pa,PG=pg,PL=pl,PW=pw,RTS[=yn6],MCI=mc,MLI=ml,]

    [P90=fv90,...,P99=fv99].


device      The element name of the terminal device being defined.  This name can be one through seven
            characters long.  The first character must be a letter; the other characters can be
            letters or digits.  If this statement is used within the set following a GROUP statement,
            the device name cannot be longer than five characters.  This is the name used by the host
            or NPU operator to monitor and control the device.  This name is required; there is no
            default value.

All other parameters are described in the text.
```

Figure 4-6. DEVICE Statement Format for Communication Lines of TIPTYPE=ASYNC

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

If you specify a CDC-defined device type, the other device definition parameters on the DEVICE or TERMDEV statement:

   Are required by the NDL processor where indicated in the text

   Have the predefined default shown in the text

   Have the predefined ranges shown in the text

If you specify a site-defined device type, the other device definition parameters on the DEVICE or TERMDEV statement:

   Are not required by the NDL processor

   Have no predefined default

   Are not checked for uniqueness

   Are not checked for compatibility with other device definition parameters on the same statement

When you specify the DT parameter, the following values are valid:

   CON   Identifies a CDC-defined console device.

   DT12   Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

If you omit the DT parameter, the default value of CON is used.

AB PARAMETER

The AB parameter specifies the character to be used to abort an output block. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

   00, 01, or 02 (characters NUL, SOH, or STX)

   3D, 7F, or 20 (characters =, DEL, or space)

   30 through 39 (characters 0 through 9)

   41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

   Any value that you declare or use by default for the BS, B1, B2, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the AB parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	AB Value Used	ASCII Character
N2741	18	CAN (CTRL and X keys)
2741	28	(

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter.

This parameter is optional. The ABL parameter has the following value range:

$$1 \leq abl \leq 7$$

The value you declare should be greater than or equal to the downline block limit (DBL parameter value) of the terminal. The host queues abl - dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory but might reduce the number of times an application program is rolled out.

If you omit the ABL parameter, a default value is used (DT of CON only). Default values depend on the value declared or used for the LSPEED parameter from the corresponding LINE or GROUP statement. The defaults are:

LSPEED Value (b/s)	ABL Value Used
$\leq$ 300	1
$\leq$ 9600	2

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether CCP should automatically connect the console device to the selected host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

NO   Indicates that CCP should not attempt automatic connection.

YES   Indicates that CCP should attempt automatic connection.

If you omit the AUTOCON parameter, the default value of NO is used for a DT of CON. If you specify this parameter without a value, the value of YES is used.

BR PARAMETER

The BR parameter determines whether CCP associates data control functions with the pressing of the break function key. The break function key is usually labeled BREAK, ATTN, INTERRUPT, or something similar.

If the break function key is not associated with data control, pressing the break function key only interrupts output. After input is completed, output is resumed at the next character.

If the break function key is associated with data control, the following occurs when the terminal user presses the break function key:

If input is in progress, the network software discards the current logical line of input (the function also associated with the character defined by the CN parameter).

If output is in progress, CCP discards all output queued for the device.

If output is in progress, or the terminal is idle, CCP sends a user break 1 message upline (the function also associated with the character defined by the B1 parameter).

After input occurs, output resumes with the next block transmitted from the host.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, the following values are recognized:

CCP   Indicates that CCP should perform the default action appropriate for the terminal class. Using this value is equivalent to omitting the parameter.

NO   Indicates that the break function key has no control functions.

YES   Indicates that the break function key has control functions.

If you omit the BR parameter or specify the value CCP, the default value of NO is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

BS PARAMETER

The BS parameter specifies the character to be used for the single-character deletion function. When the terminal user enters this character, CCP discards the preceding character transmitted by the terminal (unless the character has already been sent to the host). The character code used for the deletion function does not produce a physical backspace at the device unless the device also recognizes the code for that function.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

  00, 01, or 02 (characters NUL, SOH, or STX)

  3D, 7F, or 20 (characters =, DEL, or space)

  30 through 39 (characters 0 through 9)

  41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

  Any value that you declare or use by default for the AB, B1, B2, CN, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the BS parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the TC parameter. The default for all terminal classes except M40 is:

BS Value Used	ASCII Character
08	BS (CTRL and H keys, or <-- key, or BACKSPACE key)

Terminal class M40 has no default for this parameter.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

B1 PARAMETER

The B1 parameter specifies the character to be used as a user break 1 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 1 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 1 message as a job step interrupt from the terminal user.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

  00, 01, or 02 (characters NUL, SOH, or STX)

  3D, 7F, or 20 (characters =, DEL, or space)

  30 through 39 (characters 0 through 9)

  41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

  Any value that you declare or use by default for the AB, BS, B2, CN, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B1 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	B1 Value Used	ASCII Character
N2741	10	DLE (CTRL and P keys)
2741	3A	:

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

B2 PARAMETER

The B2 parameter specifies the character to be used as a user break 2 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 2 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 2 message as a job step termination from the terminal user.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

  00, 01, or 02 (characters NUL, SOH, or STX)

  3D, 7F, or 20 (characters =, DEL, or space)

  30 through 39 (characters 0 through 9)

  41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

  Any value that you declare or use by default for the AB, BS, B1, CN, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B2 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	B2 Value Used	ASCII Character
N2741	14	DC4 (CTRL and T keys)
2741	29	)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

CI PARAMETER

The CI parameter specifies the number of idle characters to insert in the downline data after a carriage return. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically return the carriage of the device to its left margin for the next line of output.

This parameter is optional. If you declare this parameter, you must use the reserved word CCP or a value in the following range:

$$0 \leq ci \leq 127$$

If you omit the CI parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	CI Value Used
H2000	0
M33	2
M40	1
T4014	0
2741	8
713	0
721	0
X364	0

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

CN PARAMETER

The CN parameter specifies the character to be used to abort (cancel) an input message. When the terminal user enters this character as the last character of a logical line, the network software discards the entire logical line, including any portion already transmitted from the terminal.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

    Any value that you declare or use by default for the BS, B1, B2, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CN parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	CN Value Used	ASCII Character
N2741	18	CAN (CTRL and X keys)
2741	28	(

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

CP PARAMETER

The CP parameter specifies whether CCP should make a cursor positioning response when the terminal user enters a linefeed or an end-of-line or end-of-block character, such as a CR or EOT. (The end-of-line and end-of-block characters are the current values of the ELX and ELB parameters.) The cursor positioning response to linefeed is a carriage return. The cursor positioning response used by CCP to end-of-line or end-of-block is determined by the the current settings of the ELR and EBR parameters. The default response moves the cursor to the beginning (left margin) of the next line on the screen; this is equivalent to a carriage return and linefeed operation.

The CP parameter is optional; this parameter is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following reserved words:

    CCP   Indicates that CCP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that no cursor positioning response should occur.

YES   Indicates that the cursor should be repositioned.

If you omit this parameter or specify the value of CCP, the default value of YES is used for a DT of CON.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a terminal definition command. When the terminal user enters this character as the first character on a line, CCP interprets the line as a command. Among other functions, CCP supports terminal definition commands to determine or change the values you have established for the following parameters:

AB	CP	ELR	OC	XLC
BR	CT	ELX	OP	XLTO
BS	DLC	EP	PA	XLX
B1	DLTO	HN	PG	XLY
B2	DLX	IC	PL	
CI	EBR	IN	PW	
CN	EBX	LI	TC	

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the AB, BS, B1, B2, CN, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	CT Value Used	ASCII Character
H2000	1B	ESC (CTRL and [keys or ESCAPE key)
M33	1B	ESC (CTRL and [keys or ESCAPE key)
M40	1B	ESC (CTRL and [keys or ESCAPE key)
T4014	1B	ESC (CTRL and [keys or ESCAPE key)
2741	25	%
713	1B	ESC (CTRL and [keys or ESCAPE key)
721	1B	ESC (CTRL and [keys or ESCAPE key)
X364	25	%

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between the host computer and this terminal. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following value range:

$1 \leq dbl \leq 7$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the terminal. The host queues abl - dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

If you omit the DBL parameter, a default value is used (DT of CON only). The default value depends on the value declared or used for the LSPEED parameter from the corresponding LINE or GROUP statement. The default is:

LSPEED Value (b/s)	DBL Value Used
$\leq$ 9600	1

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. Larger DBZ values cause fewer disk accesses by the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional. The DBZ parameter has the following value range:

$$1 \leq dbz \leq 2043$$

The value you declare should be chosen together with the value used for the DBL parameter.

If you omit the DBZ parameter, a default value is used (DT of CON only). Default values depend on the value declared or used for the LSPEED parameter from the corresponding LINE or GROUP statement. The defaults are:

LSPEED Value (b/s)	DBZ Value Used
$\leq$ 2400	230
$\leq$ 4800	460
$\leq$ 9600	940

These defaults optimize use of the default values for the DBL parameter.

DI PARAMETER

The DI parameter is a stand-alone keyword that specifies a device as initially enabled or disabled. An enabled device is configured and serviced as soon as the communication line becomes active. A disabled device is neither configured nor serviced when the line becomes active.

If you initially disable the device, a host or NPU operator can change the device status to enabled. This change can be made only when the line becomes active (when a call is received on a dialup line, or when communications are established on a hardwired line).

This parameter is optional. If you specify the DI parameter, you must use one of the following values:

NO   Indicates that the device is initially enabled.

YES   Indicates that the device is initially disabled.

If you specify the DI parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

DLC PARAMETER

The DLC parameter indicates the maximum number of characters that can be input in each single-message transparent mode upline message from this terminal. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent mode to normalized mode when the device transmits the indicated number of characters.

This parameter is optional; it is allowed for all terminal classes except 2741. You should not declare this parameter if you declare the XLC, XLTO, XLX, or XLY parameters. You can use the following values:

CCP

     Indicates that CCP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$1 \leq dlc \leq 4095$

     Indicates the maximum decimal number of characters that the terminal can transmit as a single transparent mode message.

If you omit this parameter or specify the value CCP, the default value of 2043 is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

DLTO PARAMETER

The DLTO parameter specifies whether a 200- to 400-millisecond timeout on the communication line ends single message transparent mode input. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent mode to normalized mode when the device stops transmitting for the indicated timeout period.

This parameter is optional; it is allowed for all terminal classes except 2741. You cannot use the DLTO parameter if you use the XLC, XLTO, XLX, or XLY parameters. When you specify the DLTO parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that a timeout does not end transparent mode input.

YES   Indicates that transparent mode input ends when a timeout occurs.

If you omit this parameter or specify the value CCP, the default value of NO is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

DLX PARAMETER

The DLX parameter indicates the hexadecimal value of the character code that should end input of a single-message transparent mode message from the device. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent mode to normalized mode when the device transmits this character.

This parameter is optional; it is allowed for all terminal classes except 2741. You cannot use the DLX parameter if you use the XLC, XLTO, XLX, or XLY parameters.

You should select transparent mode input delimiters with care. The character code you declare as the DLX value must be the code of a character that the physical device can input. If the delimiter is a character that cannot be input (either because of device hardware limitations or because of the PA parameter value declared for the device), then CCP cannot terminate transparent mode input and the device will be trapped in that mode of operation once it has begun.

The value valid for a given character depends on the character code set used by the terminal. You can use the following values:

CCP

    Indicates that CCP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \le dlx \le FF$

    Indicates the hexadecimal code that ends single message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F are not seen by CCP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, the default value of 0D is used (DT of CON only). This is the ASCII code for the carriage return character.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

EBR PARAMETER

The EBR parameter indicates the appropriate cursor positioning response whenever a block of input ends. A block of input ends when CCP receives the currently defined end-of-block character (EBX parameter value). Whether cursor positioning occurs at the end of a block is determined by the value declared or used for the CP parameter.

This parameter is optional; it is legal for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should perform the default response appropriate for the terminal class; using this value is equivalent to omitting the parameter.

CL   Indicates that CCP should perform a carriage return and linefeed operation (move the cursor left and down to the beginning of the next physical line).

CR   Indicates that CCP should perform a carriage return operation (move the cursor left to the beginning of the current physical line).

LF   Indicates that CCP should perform a linefeed operation (move the cursor down to the next physical line).

NO   Indicates that CCP should not reposition the cursor.

If you omit this parameter or specify the value CCP, the default value of CL is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

EBX PARAMETER

The EBX parameter identifies the character code that ends a block of one or more messages (one or more logical lines) from the device. When CCP detects this code in input, it forwards the last message of the block upline. The character is significant only when the terminal is operating in block input mode. If the device is operating in normalized block mode (IN parameter value of BK), CCP retains output until this code is received.

This parameter is optional; it is allowed for all terminal classes except 2741.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

    Any value that you declare or use by default for the AB, BS, B1, B2, CN, or CT parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit this parameter or specify the value CCP, the default value of 04 is used (DT of CON only). That value is the ASCII end-of-transmission (EOT) character, generated by the CTRL and D keys.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

ELR PARAMETER

The ELR parameter indicates the appropriate cursor positioning response whenever a message (logical line) of input ends. A logical line of input ends when CCP receives the currently defined end-of-line character (ELX parameter value). Whether cursor positioning occurs is determined by the value declared or used for the CP parameter.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should perform the default response appropriate for the terminal class; using this value is equivalent to omitting the parameter.

CL   Indicates that CCP should perform a carriage return and linefeed operation (move the cursor left and down to the beginning of the next physical line).

CR   Indicates that CCP should perform a carriage return operation (move the cursor left to the beginning of the current physical line).

LF   Indicates that CCP should perform a linefeed operation (move the cursor down to the next physical line).

NO   Indicates that CCP should not reposition the cursor.

If you omit this parameter or specify the value CCP, the default value of LF is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

ELX PARAMETER

The ELX parameter identifies the character code that ends a message (logical line) from the device. When CCP detects this code in input, it forwards the message upline and discards this code. If the device is operating in normalized line mode (IN parameter of KB), CCP begins output of any queued downline blocks after receiving this code.

This parameter is optional; it is allowed for all terminal classes except 2741.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the AB, BS, B1, B2, CN, or CT parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit this parameter or specify the value CCP, the default value 0D is used (DT of CON only). That value is the ASCII carriage return character.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

EP PARAMETER

The echoplex (EP) parameter causes or prevents the echoing of input characters back to the output mechanism of the device. Echoplexing is normally required when the device operates in full-duplex mode and no echoing is performed by hardware between the device input hardware and CCP.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify the EP parameter, you must use one of the following values:

CCP   Indicates that CCP should echo input if that is the appropriate default action for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that CCP should not echo input characters.

YES   Indicates that CCP should echo input characters.

If you omit the EP parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	EP Value Used
H2000	NO
M33	NO
M40	NO
T4014	NO
2741	NO
713	NO
721	NO
X364	YES

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

HD PARAMETER

The HD parameter controls whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional and is valid only for console devices.

If you specify this parameter, you must use one of the following values:

NO   Indicates that the full host availability display is not presented to the terminal user. The terminal user receives only the host status message and the prompt message.

YES   Indicates that the full host availability display is presented to the terminal user.

If you omit this parameter and only a single host is defined, the default value of NO is used. If you omit this parameter and multiple hosts are defined, the default value of YES is used.

The value of this parameter does not change when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device and any associated batch devices are connected to unless another path is selected by the terminal user.

This parameter is optional and is valid only for console devices. If you declare this parameter, you must use one of the following values:

NONE

    Indicates that no node number is specified. This has the same effect as not declaring the parameter. A value of NONE cannot be declared if AUTOCON is also specified and there are logical links to more than one host terminating in the NPU being defined.

$1 \leq hn \leq 255$

    Indicates the node number of the host that the console device and any associated batch devices are to be connected to.

The value declared for the HN parameter must be the same as the NODE value in a COUPLER statement within the same network definition. If all the COUPLER statements for all the logical links to the NPU being defined have equal HNAME parameters, the NDL processor provides a default host node. The default host node is the last coupler specified. If the HN parameter is not specified, the terminal user must select a host before a connection can be made. If the AUTOCON parameter is specified, then the HN parameter must also be specified.

The value of this parameter does not change when the terminal class is changed from the console. The terminal user can change any value that you specify.

IC PARAMETER

The IC parameter specifies whether or not the input mechanism of the device supports an ASCII DC3 code (X-OFF character) as a signal to stop input and an ASCII DC1 code (X-ON character) as a signal to resume input. CCP transmits these codes to control input flow when the device can support them.

This parameter allows you to configure a device so that input from an intelligent terminal such as a personal computer or from a cassette mechanism can be interrupted and restarted as needed without terminal user intervention. CCP sometimes needs to suspend input because the volume of network traffic has temporarily used all available storage space.

You should be careful that this parameter is appropriately defined for the device hardware that actually uses the line. These codes are used for many purposes by device manufacturers; receiving either code can have effects other than resuming or stopping input.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that CCP cannot use X-ON and X-OFF characters to control input.

YES   Indicates that CCP can use X-ON and X-OFF characters to control input.

If you omit the IC parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	IC Value Used
H2000	NO
M33	NO
M40	NO
T4014	NO
2741	NO
713	NO
721	NO
X364	YES

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

IN PARAMETER

The IN parameter identifies the input mechanism and the input message blocking of the device. (Terminal users can use a similar command that also defines whether input is in transparent mode or normalized mode; you cannot configure a device to begin accessing the network in transparent mode.)

As the input device, you can specify either:

The keyboard

The paper tape reader

As the input transmission mode, you can specify either:

Normalized line mode (one message or logical line per block)

Normalized block mode (one or more logical lines or messages collected into a block before it is transmitted. Cursor positioning at line-feed and end-of-line is not performed and output is not sent until end-of-block is reached.)

This parameter is optional; it is allowed for all terminal classes. Possible values are:

BK   Indicates keyboard input in block mode. This is not valid for TC=2741.

CCP   Indicates that CCP should use the default options appropriate for the terminal class; using this value is equivalent to omitting the parameter.

KB   Indicates keyboard input in line mode.

PT   Indicates paper tape input in block mode. This is not valid for TC=2741.

When the IN value is set to PT, CCP treats the paper tape reader as a block mode device with a fixed end-of-block character (the ASCII DC3 code, representing the X-OFF character). Whenever CCP expects input from the device, it sends an ASCII DC1 code to start the paper tape reader. This X-ON character is issued after the end of a message is output to the device and the message empties CCP's output queue for the device. If CCP receives an ASCII DC3 code (X-OFF character) after an end-of-line character code, it views that code as a signal intended to stop the tape reader; the X-OFF is discarded.

Any other DC3 code is forwarded to the host as part of the message.

If you omit this parameter or specify the value CCP, the default value of KB is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

LI PARAMETER

The LI parameter specifies the number of idle characters to insert in the downline data after a line-feed code. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically move the carriage of the device to its next line for output.

This parameter is optional. If you declare this parameter, you must use the reserved word CCP or a value in the following range:

$$0 \leq \text{li} \leq 127$$

If you omit the LI parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	LI Value Used
H2000	3
M33	1
M40	3
T4014	0
2741	1
713	0
721	0
X364	0

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

LK PARAMETER

The LK parameter specifies whether unsolicited messages from the NPU or host operator can appear at the terminal. This parameter is optional.

If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default mode appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that unsolicited messages should be delivered to the terminal as soon as received by CCP.

YES   Indicates that unsolicited messages should be discarded (locked out).

If you omit this parameter or specify the value CCP, the default value of NO is used (DT of CON only).

The value of this parameter does not revert to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

MCI PARAMETER

The MCI parameter specifies the delay after a carriage return is output. This delay is in 4-millisecond increments. This value is used by CCP to compute the number of idle characters output.

This parameter is optional. The MCI parameter has the following value range:

$$0 \leq mci \leq 250$$

Entering the terminal definition command CI to change the number of idle characters after a carriage return overrides the MCI value specified by either the application program or in the NDL. When the CI parameter and the MCI parameter are both specified in the NDL, whichever value is specified second overrides the first value. Entering the TC command to change the terminal class restores the default terminal class settings.

MLI PARAMETER

The MLI parameter specifies the delay after a line feed is output. This delay is in 4-millisecond increments. This value is used by CCP to compute the number of idle characters output.

This parameter is optional. The MLI parameter has the following value range:

$$0 \leq mli \leq 250$$

Entering the terminal definition command LI to change the number of idle characters after a line feed overrides the MLI value specified by either the application program or in the NDL. When the LI parameter and the MLI parameter are both specified in the NDL, whichever value is specified second overrides the first value. Entering the TC command to change the terminal class restores the default terminal class settings.

OC PARAMETER

The OC parameter specifies whether or not the output mechanism of the device sends an ASCII DC3 code (X-OFF character) as a signal for CCP to interrupt output and an ASCII DC1 code (X-ON character) as a signal for CCP to resume output. CCP can accept these codes for control of output flow by devices that must periodically interrupt output to perform such functions as emptying buffers to offline storage devices. DC1 and DC3 codes used in this manner are discarded by CCP.

You should only use this parameter when the device actually using the line requires it. The DC1 and DC3 codes can be sent for many purposes by terminal users; either code in input might be intended as data for the host application program.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that CCP should ignore X-ON and X-OFF characters in input; output control by the device is unnecessary.

YES   Indicates that CCP should recognize X-ON and X-OFF characters in input as output control by the device.

If you omit the OC parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	OC Value Used
H2000	NO
M33	NO
M40	NO
T4014	NO
2741	NO
713	NO
721	NO
X364	YES

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

OP PARAMETER

The OP parameter identifies the output mechanism, page width processing requirement, and page length processing requirement of the device. CCP supports these possible output mechanisms:

A console screen

A hardcopy mechanism, such as a printer

A paper tape punch

CCP performs one of these processing actions when a finite page width is reached:

Inserts the codes appropriate to return the cursor or carriage to the beginning of the next physical line

Inserts no codes, which performs no action

If the device has a page width of 0, the second choice is always used. Refer to the description under the heading PW Parameter.

CCP performs one of these processing actions when a finite page length is reached:

Inserts the codes appropriate to cause the cursor or carriage to move to the next page (clear screen, formfeed, and so forth)

Inserts no codes, which performs no action

If the device has a page length of 0, the second choice is always used. Refer to the PL Parameter description.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default options appropriate for the terminal class; using this value is equivalent to omitting the parameter.

DI   Indicates that the console display screen is used, no codes should be inserted when page width is reached (the device does its own line folding), and codes should be inserted when page length is reached.

PR   Indicates that a hardcopy printer is used, codes should be inserted when page width is reached (the device does not do its own line folding), and codes should be inserted when page length is reached.

PT   Indicates that a paper tape punch is used, codes should be inserted when page width is reached (the device does not do its own line folding), and codes should not be inserted when page length is reached.

When the OP value is set to PT, CCP sends an ASCII DC3 code (the X-OFF character) at the end-of-message with postprint format control.

If you omit this parameter or specify the value CCP, a default value is used. The default used depends on the TC parameter value declared or used; the default values are (DT of CON only):

TC Value	OP Value Used
M33	PR
2741	PR
713	DI
721	DI
X364	DI
H2000	DI
T4014	DI
M40	DI

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

PA PARAMETER

The PA parameter indicates the processing CCP should perform for the parity bit within each character byte of upline or downline data. CCP processes the parity bit of all upline character codes in one of five ways:

Ignores the input byte parity bit and sets it to zero after input. Converts the byte to an ASCII code if the device is operating in normalized mode, then forwards the byte upline to the application program. This is called zero parity processing.

Checks the input byte parity bit to ensure that it is set to zero or one, as necessary to give the byte an even number of bits; odd settings are indicated to the receiving application program as parity errors. Sets the parity bit to zero after input, converts the byte to an ASCII code if the device is operating in normalized mode, then forwards the byte upline to the application program. This is called even parity processing.

Checks the input byte parity bit to ensure that it is set to zero or one, as necessary to give the byte an odd number of bits; even settings are indicated to the receiving application program as parity errors. Sets the parity bit to zero after input, converts the byte to an ASCII code if the device is operating in normalized mode, then forwards the byte upline to the application program. This is called odd parity processing.

Ignores the input byte parity bit. If the device is operating in normalized mode, sets the parity bit to zero, converts the byte to an ASCII code, and then forwards the byte upline to the application program. If the device is operating in transparent mode, forwards the 8 bits unchanged to the application (for devices that send 8-bit bytes, this allows the application program to receive bit 7 as data). This is called no parity processing.

Ignores the input byte parity bit. If the device is operating in normalized mode, sets the parity bit to zero, converts the byte to an ASCII code, and then forwards the byte upline to the application program. If the device is operating in transparent mode, ignore the parity bit when checking for transparent mode input delimeters. Forward the 8 bits unchanged to the application in the same way as for no parity. This is called ignore parity processing.

CCP also processes the upper bit (bit 7) of all downline character code bytes in one of the following ways:

If the device is operating in normalized mode, CCP converts the lower seven bits of the byte, if necessary, to the code set used by the device. The parity bit in the output byte is set to zero, regardless of the setting of bit 7 in the downline byte. This is called zero parity processing.

If the device is operating in normalized mode, CCP converts the lower seven bits of the byte, if necessary, to the code set used by the device. The parity bit in the output byte is set to zero or one, as necessary to give the byte an even number of set bits; the setting of bit 7 in the downline byte is ignored. This is called even parity processing.

If the device is operating in normalized mode, CCP converts the lower seven bits of the byte, if necessary, to the code set used by the device. The parity bit in the output byte is set to zero or one, as necessary to give the byte an odd number of set bits; the setting of bit 7 in the downline byte is ignored. This is called odd parity processing.

If the device is operating in normalized mode, CCP converts the lower seven bits of the byte, if necessary, to the code set used by the device and the parity bit in the output byte is set to zero, regardless of the setting of bit 7 in the downline byte. If the device is operating in transparent mode, the parity bit in the output byte is unchanged from the setting of bit 7 in the downline byte (for devices that can receive 8-bit bytes, this allows the application program to use bit 7 as data). This is the method CCP uses for both no parity and ignore parity processing.

You can specify one of these input and output processing options to match the parity bit input processing performed by the device and the output processing expected by it. If the device will operate in transparent mode, you must specify either the no parity or the ignore parity option for an application program and the device to exchange hexadecimal codes with values between 7F and FF (using bit 7 as data).

No parity and ignore parity processing differ only in the way that a transparent input delimiter is recognized. For both PA=N and PA=I, all 8 bits are treated as data during transparent input and output. For PA=N, a transparent input delimiter is recognized when the 8-bit byte input from the device matches the 8-bit byte specified as the delimiter. For PA=I, a transparent input delimiter is recognized when the bottom 7 bits of the byte input from the device match the bottom 7 bits of the specified delimiter; the parity bit is ignored.

This parameter is optional. If you specify the PA parameter, you must use one of the following values:

CCP   Indicates that CCP performs the parity processing appropriate as a default for the terminal class; using this value is equivalent to omitting the parameter.

E   Indicates that CCP should perform even parity processing for both input and output bytes.

I   Indicates that CCP performs ignore parity processing for both input and output bytes.

N   Indicates that CCP should perform no parity processing for both input and output bytes.

O   Indicates that CCP should perform odd parity processing for both input and output bytes.

Z   Indicates that CCP should perform zero parity processing for both input and output bytes.

If you omit this parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the terminal class. The defaults are:

TC Value	PA Value Used
M33	E
713	E
721	E
M40	E
H2000	E
T4014	E
X364	E
2741	O

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

PG PARAMETER

The PG parameter specifies whether CCP should wait at each output page boundary for terminal user acknowledgment before it displays the next page of data. In certain situations, page waiting can occur other than at page boundaries; CCP produces a prompting message (OVER..) when this type of page waiting occurs. The user's response to page waiting is entry of a line, usually empty.

CCP views a new page as beginning at the start of each downline message. CCP calculates the length of a page from the current values of the page width and page length (see PW and PL Parameter descriptions). If the page width is infinite (PW=0), a page consists of one line less than the number of logical lines specified as the page length. If the page width is finite (PW is nonzero), a page consists of one line less than the number of physical lines specified by the page length; CCP calculates the number of physical lines by dividing each logical line into units less than or equal to the page width.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that page waiting should not occur.

YES   Indicates that page waiting should occur.

When you specify PG=YES, you should also specify a nonzero value for the PL parameter. CCP cannot perform page waiting at the boundaries of infinitely long pages (PL=0).

If you omit the PG parameter or specify a value of CCP, the default value of NO is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

PL PARAMETER

The PL parameter specifies the number of physical lines per page of output for the device. If the device is configured for page waiting (PG parameter), any message containing more lines of output than the page length will be interrupted by CCP for a page waiting response from the terminal user. The interruption occurs after line p1 - 1 is output. If the device is defined with a printer as its output mechanism (OP of PR), CCP inserts form-feed codes at page length boundaries.

This parameter is optional. If you specify the PL parameter, you must use one of the following values:

CCP

> Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

> Indicates an infinite page length. An infinite page length means that no page waiting occurs and no form-feed codes are inserted.

$8 \leq p1 \leq 255$

> Indicates the number of physical lines per page.

If you omit the PL parameter or specify the value CCP, a default value is used (DT of CON or DT of LP only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	PL Value Used
M33	0
713	24
721	30
2741	0
M40	24
H2000	27
X364	24
T4014	35

The NDL default value of the PL parameter for DT of LP is 64.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

PRI PARAMETER

The PRI parameter is a stand-alone keyword. This parameter indicates whether data to or from the device is to have traffic priority over that to or from other devices in the network. Data to or from a device that does not have traffic priority is suspended to relieve network congestion before data to or from terminals with traffic priority is suspended. Interactive devices should usually be given traffic priority over batch devices.

This parameter is optional. If you specify the PRI parameter, you must use one of the following values:

NO — Indicates that the device should not have data traffic priority.

YES — Indicates that the device should have data traffic priority.

If you declare this parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

PW PARAMETER

The PW parameter defines the number of characters per physical line of output for this device. This physical line length is also called the page width.

For console devices (DT of CON) with a printer defined as the output mechanism (OP of PR), output lines longer than pw characters are divided into lines of pw or fewer characters each. For console devices with a display defined as the output mechanism (OP of DI), output lines longer than pw characters might be divided into lines of pw or fewer characters each by the host application program; if the PW value for a device is inappropriate, loss of visual fidelity can occur (the application program might divide the data into lines that are too short or too long for the screen's capacity, instead of allowing the terminal to wrap lines when needed). An inappropriate PW value also affects line counting for page waiting; refer to the PG parameter description.

This parameter is optional. If you specify the PW parameter, you must use one of the following values:

CCP

> Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

> Indicates that the device has an infinite page width. An infinite page width means that physical line length has no effect on output format.

$20 \leq pw \leq 255$

> Indicates that the device can support physical lines no longer than the indicated decimal value.

If you omit this parameter or specify the value CCP, a default value is used. The default value depends on the terminal class. Default values for console devices (DT of CON) are:

TC Value	PW Value Used
M33	72
713	80
721	80
X364	80
2741	132
M40	80
H2000	74
T4014	74

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

P90 THROUGH P99 PARAMETERS

These ten parameters indicate the hexadecimal field value to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this device. The released version of the CDC-defined ASYNC TIP does not use any of these ten field number/field value pairs.

These parameters are optional. If you specify a value, it must be within the following range:

$0 \leq fv9i \leq FF$

If you omit these parameters, there are no default values.

RTS PARAMETER

The RTS parameter specifies whether RTS input flow control is in effect on the line.

If this parameter is selected, CCP drops the RS-232C Request to Send (RTS) signal when it needs to regulate input flow on a line. When CCP can again receive input on the line, it will raise the RTS signal and the flow of input resumes.

RTS input flow control can be used to regulate input if and only if the following two conditions are satisfied. The sending device (for example, the terminal connected to CCP) must be able to recognize one of the RS-232C signals as a sign to stop and start transmission of data. In addition, the connection between the NPU and the terminal must be wired such that RTS from CCP is received as the signal recognized by the terminal, most commonly Clear to Send (CTS).

This parameter is optional. If you specify the RTS parameter, you must use one of the following values:

NO   Indicates that the RTS signal is not raised or dropped to effect input flow control.

YES   Indicates that the RTS signal is raised and dropped to effect input flow control.

If you omit this parameter, the default value of NO is used.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input messages and if you choose a UBL value smaller than this number, the application program becomes deadlocked (it waits for blocks of data that CCP is forced to discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following value range:

$1 \leq ubl \leq 31$

The NPU queues all upline blocks. Large UBL values rapidly use up NPU memory and cause slower data transfers.

If you omit the UBL parameter, a default value is used (DT of CON only). The default value depends on the value declared or used for the LSPEED parameter from the corresponding LINE statement. The default value is:

LSPEED Value (b/s)	UBL Value Used
≤ 9600	7

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline block can contain. CCP divides each message from the device into network data blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host.

You should choose a value that allows most of the logical lines entered by the device to fit into a single network data block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional. The UBZ parameter has the following value range for console devices (DT of CON):

$0 \leq ubz \leq 2000$

For site-defined devices (DT of DT12), the following range exists:

$$0 \leq ubz \leq 2043$$

If you specify 0, CCP sends an upline block whenever it receives 100 characters, or it detects a linefeed code.

The value you declare should be chosen after considering the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes, as follows:

Value Supplied	Value in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

The NDL processor issues an informative message if it adjusts the UBZ value that you specify.

If you omit the UBZ parameter, a default value is used (DT of CON only). The default value depends on the value declared or used for the LSPEED parameter from the corresponding LINE statement. The default value is:

LSPEED Value (b/s)	UBZ Value Used
≤ 9600	100

This default optimizes use of the default values for the UBL parameter.

For the Message Control System (MCS), the upline block size must be set to 0.

The application program can change any value that you specify. The terminal user can change the value within the following range:

$$\underline{0} < ubz \leq 200$$

XBZ PARAMETER

The transmission block size (XBZ) parameter specifies the maximum number of character bytes each block sent to the terminal can contain. CCP divides downline blocks as necessary to create a block of the specified number of characters. The value you choose should be less than or equal to the size of any buffer memory within the terminal.

This parameter is optional. The XBZ parameter has the following range of values:

$$200 \leq xbz \leq 2043$$

If you omit the XBZ parameter, a default value is used (DT of CON only). The default value depends on the value declared or used for the LSPEED parameter from the corresponding LINE statement. The default value is:

LSPEED Value (b/s)	XBZ Value Used
≤ 9600	450

The application program can change any value that you specify.

XLC PARAMETER

The XLC parameter indicates the maximum number of characters that can be input in each message of multiple-message transparent mode input from this device. After the terminal user or the application program changes the device's input mode to transparent from normalized, the following occurs. When the device transmits the given number of characters, a message block is forwarded to the application and the device remains in transparent mode.

This parameter is optional; it is allowed for all terminal classes except 2741. You cannot use the XLC parameter if you use the DLC, DLTO, or DLX parameters. If specified, XLX or XLTO must also be specified. You can use the following values:

CCP

> Indicates that CCP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$1 \leq xlc \leq 4095$

> Indicates the maximum decimal number of characters that the terminal can transmit as one multiple-message transparent mode message.

If you omit this parameter or specify the value CCP, the default value of 2043 is used (DT of CON only). The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

XLTO PARAMETER

The XLTO parameter specifies whether a 200- to 400-millisecond timeout on the communication line ends multiple-message transparent mode input. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent mode to normalized mode when the device stops transmitting for the indicated timeout period.

This parameter is optional; it is allowed for all terminal classes except 2741. You cannot use this parameter if you use the DLC, DLTO, or DLX parameters.

If you specify the XLTO parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that a timeout does not end transparent mode input.

YES   Indicates that transparent mode input ends when a timeout occurs.

If you omit this parameter or specify the value CCP, no default exists. The application program or the terminal user can define a value or change any value that you specify.

XLX PARAMETER

The XLX parameter indicates the hexadecimal value of the character code that should cause upline transmission of a message in multiple-message transparent mode. If this character code is also used as the value for the XLY parameter and is input twice in succession, the device will be put back in normalized input mode as determined by the IN parameter.

This parameter is optional; it is allowed for all terminal classes except 2741. You cannot use the XLX parameter if you use the DLC, DLTO, or DLX parameters.

The value that is valid for a given character depends on the character code set used by the terminal. You can use the following values:

CCP

Indicates that CCP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \le xlx \le FF$

Indicates the hexadecimal code that ends each message within multiple-message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F are not seen by CCP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, no default exists. The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

XLY PARAMETER

The XLY parameter indicates the hexadecimal value of the character code that ends multiple-message transparent mode operation when entered following the XLX character code. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP changes back from transparent mode to normalized mode when the device transmits this character immediately after the character defined for the XLX parameter. If this character is also used for the XLX parameter, the code does not change the device's input mode unless it is input twice in succession.

You should select transparent mode input delimiters with care. The character code you declare as the XLY value must be the code of a character that the physical device can input. If the delimiter you declare is a character that cannot be input (either because of device hardware limitations or because of the PA parameter value declared for the device), then CCP cannot terminate transparent mode input and the device will be trapped in that mode of operation once it has begun.

This parameter is optional; it is allowed for all terminal classes except 2741. You should not declare this parameter if you declare the DLC, DLTO, or DLX parameters. If you use the XLY parameter, you must also use the XLX parameter.

The value that is valid for a given character depends on the character code set used by the terminal. You can use the following values:

CCP

Indicates that CCP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \le xly \le FF$

Indicates the hexadecimal code that ends multiple-message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F are not seen by CCP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, and XLX is also specified, the default value is the code specified for XLX. Otherwise, no default exists. The value of this parameter reverts to the default of no value when the terminal class is changed from the console or by the application program. The application program or the terminal user can change any value that you specify.

BUFFERING OF DATA

It is desirable to maintain double buffering of data for each connection in the network. By using double buffering data will always be available to be sent over the communications line. Setting the ABL parameter to 2 and the DBL parameter to 1 will generally result in double buffering.

CCP buffer usage is optimized when downline block size follows the following relationship:

$$DBZ = 113 + n*118$$

where n is an integer and n + 1 is the number of buffers required in CCP to hold the block. This relationship is of most use for application programs that can control the size of the downline blocks they send to the network. Since it is necessary for a block to contain an integral number of logical lines, the lower bound of DBZ should not be smaller than the largest allowable PW for a device.

Default values for the ABL, DBL, and DBZ parameters are listed in table 4-2.

TABLE 4-2. ABL, DBL, AND DBZ DEFAULTS

Line Speed	ABL	DBL	DBZ
110	1	1	230
300	1	1	230
600	2	1	230
1200	2	1	230
2400	2	1	230
4800	2	1	460
9600	2	1	940

The parameters required and the values allowed on the LINE, TERMINAL, TERMDEV, and DEVICE statements depend upon the line protocol used. This section describes the form of a statement applicable to X.25 protocol lines. The first part of this section describes only terminal definitions. Application definitions are described after terminal definitions.

X.25 communication lines are those using the line types defined in table 5-1. Such lines are serviced by the CDC-supplied X.25 TIP and must be defined as fixed-configuration lines. If you use the site-defined line type, the X.25 TIP is presumed to be modified by your site (if necessary) to support either a standard communication line adapter or a specially built communication line adapter.

LINE DEFINITION

You configure terminals on an X.25 communication line by using the following statements:

One LINE statement that defines the line

A single TERMINAL or TERMDEV statement for all switched virtual circuits of the X25 subTIP on the line

A single TERMINAL or TERMDEV statement for all switched virtual circuits of the PAD subTIP on the line

One DEVICE statement for all switched virtual circuits of the PAD, X25, or USER subTIP on the line, unless a TERMDEV statement has been used

LINE STATEMENT PARAMETERS

Each LINE statement defines one communication line between an X.25 network or device and the NPU. There must be one LINE statement for each CLA port on the NPU that supports X.25 access.

Figure 5-1 shows the format of the LINE statement and the valid parameter values for X.25 communication lines. Table 5-1 shows permitted line type values for the LTYPE parameter.

As an example, the statement

```
LN11A: LINE PORT=11,LTYPE=H1,TIPTYPE=X25,
        DFL=32,FRAME=1,RTIME=1,PSN=TELENET.
```

defines a high-level data link control (HDLC) line connected to the TELENET X.25 packet-switching network.

TABLE 5-1. X.25 LINE TYPE DEFINITIONS

LTYPE Value	Transmission Mode	Transmission Operation	Circuit Type	Modem Type†	CLA Type	Maximum Speed, Bits per Second	Carrier Type
H1	Bit-oriented synchronous HDLC	Full-duplex	Dedicated (Hardwired)	RS232C, Bell 201B compatible, V.35 Standard compatible	2563-1 / 2563-2	19200 / 56000	Constant
H2 (For site-defined use)	HDLC	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned

†Modem types supported by each type of CLA may differ.

line: LINE,PORT=port,LTYPE=ltype,TIPTYPE=tiptype,[DI[=yn1],]

   DFL=dfl,FRAME=frame,RTIME=timer,PSN=psn[,RCOUNT=count,]

   [NSVC=svcirc,DCE[=yn2],DTEA=locadr,IMDISC[=yn3],LCN=lcn,AL=acclev,RC[=yn4],P90=fv90,...,P99=fv99].

line   The element name to be assigned to the communication line being defined. This name is
          required; there is no default value.

port   The hexadecimal number (1 ≤ port ≤ FE) of the port to which this communication line connects
          on the NPU currently being defined. Within an NDL program, the port number is independent of
          the number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE (254
          decimal). However, we strongly recommend that you assign port numbers consecutively, starting
          with 1. All values declared for PORT parameters must be unique within the current network
          definition of each NPU. The value specified for port cannot be the same as the number
          declared for a port in any other LINE or TRUNK statement (or within a GROUP statement
          expansion) for this NPU and cannot be lower than the highest port number used by a trunk.
          This number is required; there is no default value.

ltype   A reserved word value that identifies the type of communication line adapter/modem/circuit
          combination that is used on this line. This word is required; there is no default value. The
          legal words for this value declaration are described in table 5-1.

tiptype   A reserved word value that identifies the type of CCP Terminal Interface Program required for
          the terminals on this line. This word is required; there is no default value. The legal
          value for this declaration is:

            X25   X.25 protocol TIP is required.

yn1   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this
          parameter specifies the status assigned to this communication line at network initiation.
          When DI or DI=YES is specified, the line is disabled at network initiation and cannot be used
          until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is given an
          initial status of enabled unless the NOP or HOP specifies otherwise.

dfl   The maximum decimal number (16 ≤ dfl ≤ 1024) of 8-bit characters or octets in a data packet.
          The parameter must be identical to the number of octets in a data packet contracted from the
          packet-switching network. This value is required; there is no default value.

frame   The frame window (1 ≤ frame ≤ 7); the maximum number of outstanding unacknowledged packets
          between the network processing unit and the X.25 network (this is the K parameter defined by
          the X.25 protocol standard). The parameter must be identical to the frame window size
          contracted from the packet-switching network. This value is required; there is no default
          value.

timer   The decimal retransmission timing period (1 ≤ timer ≤ 25500) in milliseconds; the time that
          must elapse before retransmission of an unackowledged X.25 data frame is attempted (this is
          the T1 parameter defined by the X.25 protocol standard). This value is required; there is no
          default value. NDLP rounds the value specified for timer to the next multiple of 100.

count   The decimal retransmission count (1 ≤ count ≤ 15); the number of times retransmission of an
          unacknowledged X.25 data frame is attempted (this is the N2 parameter defined by the X.25
          protocol standard). This parameter is optional. When count is not specified, the default
          value of 15 is used.

svcirc   The decimal number (0 ≤ svcirc ≤ 255) of available switched virtual circuits (SVC) for this
          line. This parameter is optional; the default is 0. The parameter must be identical to the
          number of switched virtual circuits contracted from the packet-switching network and must
          conform to the following rules:

          The total number of virtual circuits defined (by the NCIR parameter on the TERMINAL or TERMDEV
          statement) for each STIP must be ≤ svcirc.

psn   A reserved word value identifying the packet-switching network to which the line is con-
          nected. This parameter is required; there is no default value. The following values are
          recognized:

            CDSN   Packet-switching network is compatible with the commercial CDSN network.

Figure 5-1. X.25 LINE Statement Format (Sheet 1 of 2)

	C120	Packet-switching network is compatible with the CDC CYBER 120.

C120   Packet-switching network is compatible with the CDC CYBER 120.

DATAPAC   Packet-switching network is compatible with the commercial DATAPAC network.

PSN253   Site-defined packet-switching network type, identified by the number 253 within
          the network software.

PSN254   Site-defined packet-switching network type, identified by the number 254 within
          the network software.

PSN255   Site-defined packet-switching network type, identified by the number 255 within
          the network software.

TELENET   Packet-switching network is compatible with the commercial TELENET network.

TRNSPAC   Packet-switching network is compatible with the commercial TRANSPAC network.

TYMNET   Packet-switching network is compatible with the commercial TYMNET network.

UNINET   Packet-switching network is compatible with the commercial UNINET network.

Commercial packet-switching networks do not have identical interface requirements. CDC supports as general an X.25 interface as possible. As of the PSR level shown on the record of revision page, CDC code supported all of the commercial networks listed above. TELENET and TYMNET have certified CDC support; when major changes occur in either of these packet-switching networks, CDC verifies continued certification and releases any needed support code as soon as feasible. Users of the other commercial networks should contact their local sales offices if questions of continued compatibility arise.

yn2   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter tells the NPU what role to play in the X.25 protocols. If DCE or DCE=YES is specified, the NPU must act as data circuit terminating equipment (DCE). If DCE is omitted or DCE=NO is specified, the NPU must act as data terminal equipment (DTE).

locadr   The decimal address ($0 \le$ locadr ≤ 99) assigned to the local end of the X.25 link. This value specifies the address by which the X.25 protocol identifies calls to and from the connected NPU. The value declared will be the calling DTE address for outgoing call requests when DCE=YES is specified. This parameter is optional; if DTEA is specified, DCE also must be specified. There is no default value.

yn3   An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on the line is disconnected immediately after the terminal has logged out (disconnected) from the host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is disconnected only after a two minute timer expires.

lcn   The logical channel number ($0 \le$ lcn ≤ 255) which corresponds to the lowest numbered channel that a DTE CCP can use to make an outgoing application-to-application call. If CCP is a DCE, it corresponds to the highest numbered virtual circuit CCP can use to make outgoing application-to-application calls. This value must not exceed the total number of switched virtual circuits specified by the NSVC parameter. This parameter is optional; the default is 0. All NSVC circuits can be used.

acclev   The decimal access level limit ($0 \le$ acclev ≤ 7) for the line. The lowest access level is 0 (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level limit is associated with the line (equivalent to specifying AL=0).

yn4   An optional reserved word value (YES or NO) which specifies whether the reconfiguration indicator is enabled for this line. When RC or RC=YES is specified, the terminal characteristics are reset to their original NDL values (or to a default if no NDL values were specified) when the terminal disconnects from a host. When RC is omitted or RC=NO is specified, the reconfiguration indicator is not enabled for this line.

fv9i   The hexadecimal field value ($0 \le$ fv9i $\le$ FF) to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this line. The released version of the CDC-written X.25 TIP does not use any of these ten field number/field value pairs.

Figure 5-1. X.25 LINE Statement Format (Sheet 2 of 2)

PAD AND USER TERMINAL DEFINITIONS

On an X.25 communication line, a terminal definition is the first portion of a virtual circuit definition. An X.25 terminal and device definition, however, does not directly represent a piece of physical equipment. Each X.25 terminal and device definition represents one or more switched virtual circuits. Each virtual circuit can service one terminal device.

Switched virtual circuits are comparable to terminal devices using dialup lines (the characteristics of the device using the line can change from use to use of the line). You specify only one terminal device definition for the terminals that use the line. As each switched virtual circuit becomes active, CCP creates a separate device definition for it, creating an element name for each definition. This device name consists of the one to five characters you declare, plus an ordinal consisting of the switched virtual circuit number.

You must provide one TERMINAL or TERMDEV statement for all packet assembly/disassembly (PAD) terminals that can access the communication line. Figures 5-2 and 5-3 present the formats of these statements for terminals on X.25 lines.

```
TERMINAL[,STIP=stiptyp,TC=trmclas,]

    [CSET=charset,RIC=yn1,]

    [W=pacwndw,NCIR=numcir,]

    [NEN=encir,PAD=string,COLLECT[=yn2]].

Parameters are described in the text.
```

Figure 5-2. TERMINAL Statement Format
for Communication Lines of TIPTYPE=X25

If any switched virtual circuits are defined, the LINE statement must specify the total number of circuits available from the packet-switching network via that line (NSVC parameter). When the line supports switched virtual circuits, you can declare only one TERMINAL and DEVICE or one TERMDEV statement for each subTIPtype (STIP parameter) on the line. You can use the TERMINAL or TERMDEV statement NCIR parameter to define more than one circuit with that statement. The NCIR parameter determines the highest ordinal number used to generate a unique element name for each device using the circuit.

The following parameters provide the terminal definition for the line:

COLLECT	NEN	STIP
CSET	PAD	TC
NCIR	RIC	W

The STIP and TC parameters are described first because they are related and affect the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

STIP PARAMETER

The STIP parameter specifies the packet protocol required to support the terminal definition. This parameter is optional if the TC parameter does not specify a site-defined terminal class. If STIP is omitted, the TC parameter must be specified and cannot be CCP or TC28 through TC31.

When you declare a value for STIP, it must be one of the following reserved words:

PAD   Identifies a terminal that is connected via the asynchronous stop-start (X.29) packet assembly/disassembly protocol.

USER   Identifies a terminal that uses a site-defined packet protocol.

```
device:  TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,RIC=yn1,W=pacwndw,]

    [NCIR=numcir,PAD=string,NEN=encir,COLLECT[=yn2],DT=devtyp,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

    [DBL=dwnblim,UBL=upblim,AUTOCON[=yn3],PRI[=yn4],HN=node,HD[=yn5],LK=yn6,BR=br,BS=bs,]

    [B1=b1,B2=b2,CI=ci,CN=cn,CP=cp,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,ELR=elr,]

    [ELX=elx,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,IN=in,LI=li,OC=oc,OP=op,PA=pa,]

    [PG=pg,PL=pl,PW=pw,MCI=mc,MLI=ml,P90=fv90,...,P99=fv99].

device      The element name of the terminal device being defined. This name can be one through five
            characters long. The first character must be a letter; the other characters can be letters
            or digits. This name concatenated with the virtual circuit number is used by the host or
            NPU operator to monitor and control the device. This name is required; there is no default
            value.

All other parameters are described in the text.
```

Figure 5-3. TERMDEV Statement Format for Communication Lines of TIPTYPE=X25

If you omit STIP, you must declare the terminal class. The defaults are:

TC Value	STIP Value Used
M33, 713, 721, M40, H2000, X364, T4014	PAD

TC PARAMETER

The TC parameter specifies the terminal class appropriate for terminal connections. Supported devices are grouped into terminal classes, according to their hardware characteristics.

Each CDC-defined terminal class has the default characteristics associated with an archetype terminal. Each CDC-defined terminal class has a range of possible characteristics. These ranges determine the values legal for parameters you specify on the TERMDEV or DEVICE statements.

The mnemonics identifying the terminal classes are derived from the identification of the archetype terminals defining the classes, with the following exceptions: you can use TC28, TC29, TC30, and TC31 to identify site-defined terminal classes with no default characteristics, corresponding to reserved Network Access Method (NAM) terminal class numbers.

Because the network cannot recognize differences among some terminal classes, a default terminal class exists for each subcategory of the TIP type (the subTIPtype, defined by the STIP parameter of the TERMINAL or TERMDEV statement); these defaults can be used when you declare a STIP parameter.

Because the characteristics of terminals in site-defined classes are unknown, TERMINAL, TERMDEV, or DEVICE statement parameter value ranges are checked against those valid for the STIP value specified. Any value declared for a terminal or device parameter is valid if the parameter and value conform to requirements of that subTIPtype.

The TC parameter is optional and is allowed only for PAD or USER subTIPs. If TC is omitted, the STIP parameter must be specified. If TC is specified as CCP or TC28 through TC31, the STIP parameter must be specified. If STIP is omitted, the TC parameter must be specified and cannot be CCP or TC28 through TC31.

When you declare a value for TC, you must use one of the following words:

CCP   Specifies that CCP is to provide the default terminal class appropriate for the subTIPtype; using this value is equivalent to omitting the parameter.

H2000   Identifies a terminal device compatible with Hazeltine 2000 series equipment emulating a teletypewriter.

M33   Identifies a terminal device compatible with Teletype Model 30 series equipment.

M40   Identifies a terminal device compatible with Teletype Model 40-2 equipment.

TC28   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.

TC29   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.

TC30   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.

TC31   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

T4014   Identifies a terminal device compatible with Tektronix 4014 series equipment emulating a teletypewriter.

X364   Identifies a terminal device compatible with the ANSI X3.64 standard (DEC VT100 or CDC 722-30).

713   Identifies a terminal device compatible with CDC Model 713, 722-10/20, 751, 752, or 756 series equipment.

721   Identifies a terminal device compatible with CDC Model 721 equipment.

If the TC parameter is specified with a value of TC28, TC29, TC30, or TC31, then CCP must also be modified to accept these values. If the appropriate support code for TC28, TC29, TC30, or TC31 is not added to CCP, these values will be considered invalid by CCP.

If you omit the TC parameter or use the value CCP, the default is selected according to the STIP value declared. The default is:

STIP Value	TC Value Used
PAD	M33

COLLECT PARAMETER

The COLLECT parameter specifies whether CCP accepts charges for incoming calls from X.25 terminals connected through the packet assembly/disassembly service. This stand-alone parameter is optional.

The COLLECT parameter can have the following values:

NO   CCP rejects any incoming call when the X.25 network indicates that charges will occur.

YES   CCP accepts charges from the X.25 network.

If you specify the COLLECT parameter without a value, the value of YES is used. If you omit the COLLECT parameter, the default value of NO is used.

CSET PARAMETER

The CSET parameter specifies the code and character set of the terminal. The CSET parameter is optional.

When you declare a value for CSET, you must use one of the following words:

ASCII ASCII code and character set

CSET15 Site-defined code and character set, identified within the network software as character set number 15

If you omit CSET, the default value of ASCII is used.

NCIR PARAMETER

The NCIR parameter specifies the decimal number of virtual circuits of the same subTIPtype you are defining. This parameter is optional.

If you specify this parameter, the value you use must be less than or equal to the NSVC value declared or used on the corresponding LINE statement. The NCIR parameter has the following range of values:

$$1 \leq numcir \leq 255$$

If you omit this parameter, the default value of 1 is used.

NEN PARAMETER

The NEN parameter specifies the decimal number of virtual circuits of the same subTIPtype that are intially enabled. This parameter is optional.

If you specify this parameter, the value you use must be less than or equal to the NCIR value declared or used on the same TERMINAL or TERMDEV statement. The NEN parameter has the following range of values:

$$1 \leq encir \leq 255$$

If you omit this parameter, the default value of 1 is used.

PAD PARAMETER

The PAD parameter specifies a string of 4 to 64 hexadecimal digits representing the ASCII equivalent of PAD parameter fields. Substrings should consist of groups of 4 hexadecimal digits (2 ASCII characters). CCP uses the value to initialize PAD parameters when the terminal is connected.

The PAD parameter is optional. There is no default value.

RIC PARAMETER

The RIC parameter indicates that the terminal has restricted interactive capabilities. This parameter is optional.

A terminal can have restricted interactive capabilities for many reasons; for example, perhaps no console device actually exists and interactive dialog must occur through a site-defined device.

The effect this parameter has depends on the application program the terminal uses. For example, RBF does not output the READY prompt and does not require a GO command to enable any defined passive devices when RIC is specified.

You can use the following values for this parameter:

NO   Indicates that the terminal has full interactive capabilities.

YES Indicates that the terminal has limited interactive capablities.

If you omit the RIC parameter, the default value of NO is used.

W PARAMETER

The W parameter specifies the default packet level window size for the virtual circuit you are defining. This parameter is optional.

The packet level window is the maximum number of unacknowledged packets you allow CCP to send before it suspends further output to the terminal using the virtual circuit. The parameter must be identical to the packet level window size contracted from the packet-switching network. This parameter has the following range of values:

$$1 \leq pacwndw \leq 7$$

If you omit the W parameter, the default value of 2 is used.

PAD AND USER DEVICE DEFINITIONS

You must provide one TERMDEV or DEVICE statement for:

All switched virtual circuit application-to-application connections

All switched virtual circuit terminals

that can access the communication line through the terminal. Each TERMINAL statement can have only one DEVICE statement. Figures 5-3 and 5-4 present the formats of the TERMDEV and DEVICE statements for terminals on X.25 lines.

The following parameters provide the definition for a CDC-defined X.25 terminal device:

ABL	DBZ	HN	PL
AUTOCON	DLC	IC	PRI
BR	DLTO	IN	PW
BS	DLX	LI	P90 through P99
B1	DT	LK	UBL
B2	EBR	MCI	UBZ
CI	EBX	MLI	XLC
CN	ELR	OC	XLTO
CP	ELX	OP	XLX
CT	EP	PA	XLY
DBL	HD	PG	

The DT parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

```
device:  DEVICE[,DT=devtyp,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

     [AUTOCON[=yn1],PRI[=yn2],HN=node,HD[=yn3],LK=yn4,BR=br,BS=bs,B1=b1,B2=b2,CI=ci,CN=cn,]

     [CP=cp,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,ELX=elx,ELR=elr,EP=ep,]

     [XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,IN=in,LI=li,OC=oc,OP=op,PA=pa,PG=pg,PL=pl,PW=pw,]

     [MCI=mc,MLI=ml,P90=fv90,...,P99=fv99].


device        The element name of the terminal device being defined. This name can be one through five
              characters long. The first character must be a letter; the other characters can be letters
              or digits. This name concatenated with the virtual circuit number is used by the host or
              NPU operator to monitor and control the device. This name is required; there is no default
              value.

All other parameters are described in the text.
```

Figure 5-4. DEVICE Statement Format for Communication Lines of TIPTYPE=X25

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

If you specify a CDC-defined device type and the terminal has a STIP of PAD, the other device definition parameters on the DEVICE or TERMDEV statement:

Are required by the NDL processor where indicated in the text

Are checked for compatibility and uniqueness

Have the predefined ranges shown in the text

If you specify a site-defined device type or the terminal has a STIP of USER, the other device definition parameters on the DEVICE statement:

Are not required by the NDL processor

Have no predefined default

Are checked for multiple definitions of the same parameter

Are not checked for compatibility with other device definition parameters on the same statement

When you declare the DT parameter, the following values are valid:

CON   Identifies a CDC-defined console device (for PAD subTIP only).

DT12   Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

If you omit the DT parameter and STIP=PAD, the value of CON is used.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this virtual circuit. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter. Refer to the description under Buffering of Data at the end of this section.

This parameter is optional. The ABL parameter has the following range of values:

$1 \leq abl \leq 7$

The value you declare should be greater than or equal to the downline block limit (DBC parameter value) of the terminal. The host queues abl – dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory but might reduce the number of times an application program is rolled out.

If you omit the ABL parameter, a default value is used (DT of CON only). The default is 2.

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether CCP should automatically connect the console device to the selected host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

NO   Indicates that CCP should not attempt automatic connection.

YES   Indicates that CCP should attempt auto-
        matic connection.

If you omit the AUTOCON parameter, the default
value of NO is used for a DT of CON. If you specify
this parameter without a value, the value of YES is
used.

BR PARAMETER

The BR parameter determines whether CCP associates
data control functions with the pressing of the
break function key. The break function key is
usually labeled BREAK, ATTN, INTERRUPT, or some-
thing similar.

When the break function key is not associated with
data control, pressing the break function key only
interrupts output. After input is completed, out-
put will be resumed at the next character.

When the break function key is associated with data
control, the following occurs when the terminal
user presses the key:

    If output is in progress, CCP discards all out-
    put queued for the device.

    If the terminal is idle or output is in pro-
    gress, CCP sends a user break 1 message upline
    (the function also associated with the character
    defined by the B1 parameter).

After input occurs, output resumes with the next
block transmitted from the host.

The BR parameter is optional. If you specify this
parameter, the following values are recognized:

    CCP   Indicates that CCP should perform the
            default action appropriate for the
            terminal class. Using this value is
            equivalent to omitting the parameter.

    NO   Indicates that the break function key
            has no control functions.

    YES   Indicates that the break function key
            has control functions.

If you omit the BR parameter or specify the value
CCP, the default value of NO is used (DT of CON
only).

The value of this parameter reverts to the default
value when the terminal class is changed from the
console or changed by the application program. The
application program or terminal user can also change
the value you specify.

BS PARAMETER

The BS parameter specifies the character to be used
for the single-character deletion function. When
the terminal user enters this character, CCP
discards the preceding character transmitted by the
terminal (unless the character has already been
sent to the host). The character code used for the
deletion function does not produce a physical
backspace at the device unless the device also
recognizes the code for that function.

This parameter is optional. If you specify a value,
it must be either the reserved word CCP or a hexa-
decimal value for the 7-bit ASCII code that
represents the character to be used (table A-2 in
appendix A lists all of the codes in the 7-bit
ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic
    characters A through Z or a through z)

    Any value that you declare or use by default
    for the B1, B2, CN, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character
set, you should choose a character that has a unique
translation to a character in the 7-bit ASCII code
set.

If you omit the BS parameter or specify the value
CCP, the default value is used (DT of CON only).
The default value depends on the value declared or
used for the TC parameter. The default for all
CDC-defined terminal classes except M40 is:

BS Value Used	ASCII Character
08	BS (CTRL and H keys, or <-- key, or BACKSPACE key)

Terminal class M40 has no default character for this
function.

The value of this parameter reverts to the default
value when the terminal class is changed from the
console or changed by the application program. The
application program or terminal user can also change
the value you specify.

B1 PARAMETER

The B1 parameter specifies the character to be used
as a user break 1 indicator. When the terminal user
enters this character as the only character on a
line, CCP discards the block of data being trans-
mitted to the terminal and sends a user break 1
message upline to the connected application program.

The effect this parameter has depends on the
application program. For example, IAF interprets
the user break 1 message as a job step interrupt
from the terminal user.

This parameter is optional. If you specify a value,
it must be either the reserved word CCP or a hexa-
decimal value for the 7-bit ASCII code that
represents the character to be used (table A-2 in
appendix A lists all of the codes in the 7-bit
ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the BS, B2, CN, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B1 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The default is:

STIP Value	B1 Value Used	ASCII Character
PAD	10	DLE (CTRL and P keys)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

B2 PARAMETER

The B2 parameter specifies the character to be used as a user break 2 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 2 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 2 message as a job step termination from the terminal user.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the BS, B1, CN, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

STIP Value	B2 Value Used	ASCII Character
PAD	14	DC4 (CTRL and T keys)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

CI PARAMETER

The CI parameter specifies the number of idle characters to insert in the downline data after a carriage return. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically return the carriage of the device to its left margin for the next line of output.

This parameter is optional. If you declare this parameter, you must use the reserved word CCP or a value in the following range:

$$0 \le ci \le 127$$

If you omit the CI parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	CI Value Used
H2000	0
M33	2
M40	1
T4014	0
713	0
721	0
X364	0

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

CN PARAMETER

The CN parameter specifies the character to be used to abort (cancel) an input message. When the terminal user enters this character as the last character on a line, the network software discards the last logical line transmitted from the terminal.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the BS, B1, B2, CT, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CN parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The default is:

STIP Value	CN Value Used	ASCII Character
PAD	18	CAN (CTRL and X keys)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

CP PARAMETER

The CP parameter specifies whether CCP should make a cursor positioning response when the terminal user enters a packet-forwarding signal. A packet-forwarding signal is usually a carriage return but can be set by the terminal user to other values. CCP can perform cursor positioning only on receipt of a packet sequence.

The cursor positioning response used by CCP is determined by the the current settings of the EBR parameter. The default response for terminals that are using X.25 lines is no response.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP	Indicates that CCP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.
NO	Indicates that no cursor positioning response should occur.
YES	Indicates that the cursor should be repositioned.

If you omit this parameter or specify the value of CCP, the default value of NO is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a terminal definition command. When the terminal user enters this character as the first character on a line, CCP interprets the line as a command. Among other functions, CCP supports terminal definition commands to determine or change the values you have established for the following parameters:

BR	DLC	IC	PW
BS	DLTO	IN	TC
B1	DLX	LI	XLC
B2	EBR	OC	XLTO
CI	EBX	OP	XLX
CN	ELX	PA	XLY
CP	EP	PG	
CT	HN	PL	

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the BS, B1, B2, CN, EBX, or ELX parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The default is:

STIP Value	CT Value Used	ASCII Character
PAD	1B	ESC (CTRL and [keys or ESCAPE key)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between the host computer and this virtual circuit. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following value range:

$$1 \leq dbl \leq 7$$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the virtual circuit. The host queues abl - dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

If you omit the DBL parameter, a default value is used (DT of CON only). The default value is 2.

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional. The DBZ parameter has the following range of values:

$1 \leq dbz \leq 2043$

The value you declare should be chosen together with the value used for the DBL parameter.

If you omit the DBZ parameter, a default value is used (DT of CON only). The default value is 225. This default optimizes use of the default value for the DBL parameter.

DLC PARAMETER

The DLC parameter indicates the maximum number of characters that can be input in each single-message transparent mode upline message from this terminal. After the terminal user or application program changes the device´s input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent to normalized mode when the device transmits the indicated number of characters.

This parameter is optional. You cannot use the DLC parameter if you use the XLC, XLTO, XLX, or XLY parameters. If you specify the DLC parameter, you must use one of the following values:

CCP

Indicates that CCP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$1 \leq dlc \leq 4095$

Indicates the maximum decimal number of characters that the terminal can transmit as a single transparent mode message.

If you omit this parameter or specify the value CCP, the default value of 2043 is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

DLTO PARAMETER

The DLTO parameter specifies whether the end-of-packet sequence ends single message transparent mode input. After the terminal user or application program changes the device´s input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent to normalized mode when the device transmits the end-of-packet sequence.

This parameter is optional. You cannot use the DLTO parameter if you use the XLC, XLTO, XLX, or XLY parameters. If you specify the DLTO parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that end-of-packet sequence does not end transparent mode input.

YES   Indicates that transparent mode input ends at end-of-packet sequence.

If you omit this parameter or specify the value CCP, the default value of NO is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

DLX PARAMETER

The DLX parameter indicates the hexadecimal value of the character code that should end input of a single-message transparent mode message from the device. After the terminal user or application program changes the device´s input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent to normalized mode when the device transmits this character.

This parameter is optional. You cannot use the DLX parameter if you use the XLC, XLTO, XLX, or XLY parameters.

The value that is valid for a given character depends on the character code set used by the terminal. You can use the following values:

CCP

Indicates that CCP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \leq dlx \leq FF$

Indicates the hexadecimal code that ends single message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F are not seen by CCP unless the eighth (parity) bit of each input byte is defined as data.

You should select transparent mode input delimiters with care. The character code you declare as the DLX value must be the code of a character that the physical device can input. If the delimiter is a character that cannot be input (either because of terminal hardware limitations or because of the PA parameter value declared for the device), then the Terminal Interface Program cannot terminate transparent mode input and the terminal will be trapped in that mode of operation once it has begun.

If you omit this parameter or specify the value CCP, the default value of 0D is used (DT of CON only). This is the ASCII code for the carriage return character.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

EBR PARAMETER

The EBR parameter indicates the appropriate cursor positioning response whenever a block of input ends. A block of input ends when CCP detects the end-of-packet indicator (M bit value of zero). Whether cursor positioning occurs is determined by the value declared or used for the CP parameter.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should perform the default response appropriate for the terminal class; using this value is equivalent to omitting the parameter.

CL   Indicates that CCP should perform a carriage return and linefeed operation (move the cursor left and down to the beginning of the next physical line).

CR   Indicates that CCP should perform a carriage return operation (move the cursor left to the beginning of the current physical line).

LF   Indicates that CCP should perform a linefeed operation (move the cursor down to the next physical line).

NO   Indicates that CCP should not reposition the cursor.

If you omit this parameter or specify the value CCP, the default value of CL is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program.

EBX PARAMETER

The EBX parameter identifies the character code that ends a block of one or more messages is the packet-forwarding signal from the device. When this code occurs as the last character in a packet sequence, it is treated as the end-of-block

character and not as part of normalized mode data; CCP discards it. This parameter is optional. It is significant only when the terminal is in block input mode.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 7E (all graphic characters)

    Any value that you declare or use by default for the BS, B1, B2, CN, or CT parameters

If you omit this parameter or specify the value CCP, the default value 04 is used (DT of CON only). That value is the ASCII end-of-transmission character (EOT) generated by the CTRL and D keys.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

ELR PARAMETER

The ELR parameter indicates the appropriate cursor positioning response whenever a message (logical line) of input ends. A logical line of input ends when CCP receives the currently defined end-of-line character (ELX parameter value). Whether cursor positioning occurs is determined by the value declared or used for the CP parameter.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP performs the default response appropriate for the terminal class; using this value is equivalent to omitting the parameter.

CL   Indicates that CCP performs a carriage return and linefeed operation (move the cursor left and down to the beginning of the next physical line).

CR   Indicates that CCP performs a carriage return operation (move the cursor left to the beginning of the current physical line).

LF   Indicates that CCP performs a linefeed operation (move the cursor down to the next physical line).

NO   Indicates that CCP does not reposition the cursor.

If you omit this parameter or specify the value CCP, the default value of LF is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

ELX PARAMETER

The ELX parameter identifies the character code that ends a message (logical line) to or from the device. When CCP detects this code in input, it forwards the message upline.

This parameter is optional. You cannot declare the ELX parameter if you declare the ELO parameter for the same device.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 7E (all graphic characters)

    Any value that you declare or use by default for the BS, B1, B2, CN, or CT parameters

If you omit this parameter or specify the value CCP, the default value of OD is used (DT of CON only). That value is the ASCII carriage return character.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

EP PARAMETER

The echoplex (EP) parameter causes the echoing of input characters back to the output mechanism of the device. Echoplexing is normally required when the device operates in full-duplex mode and no echoing is performed by hardware between the device input hardware and the PAD.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify the EP parameter, you must use one of the following values:

    CCP   Indicates that CCP should notify the PAD
            to echo input if that is the appropriate
            action for the terminal class; using
            this value is equivalent to omitting the
            parameter.

    NO   Indicates that CCP should not notify the
            PAD to echo input characters.

    YES   Indicates that CCP should notify the PAD
            to echo input characters.

If you omit the EP parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	EP Value Used
H2000	NO
M33	NO
M40	NO
T4014	NO
2741	NO
713	NO
721	NO
X364	YES

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value you specify.

HD PARAMETER

The HD parameter controls whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional.

If you specify this parameter, you must use one of the following values:

    NO   Indicates that the full host avail-
            ability display is not presented to the
            terminal user. The terminal user
            receives only the host status message
            and the prompt message.

    YES   Indicates that the full host avail-
            ability display is presented to the
            terminal user.

If you omit this parameter and only a single host is defined, the default value of NO is used. If you omit this parameter and multiple hosts are defined, the default value of YES is used.

The value of this parameter does not change when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change the value you specify.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device is connected to unless another path is selected by the terminal user.

This parameter is optional. If you declare this parameter, you must use one of the following values:

    NONE

        Indicates that no node number is specified.
        This has the same effect as not declaring
        the parameter. A value of NONE cannot be
        declared if AUTOCON is also specified and
        there are logical links to more than one
        host terminating in the NPU being defined.

$$1 \leq hn \leq 255$$

Indicates the node number of the host that the console device and any associated batch devices are to be connected to.

The value declared for the HN parameter must be the same as the NODE value in a COUPLER statement within the same network definition. If all the COUPLER statements for all the logical links to the NPU being defined have equal HNAME parameters, the NDL processor provides a default host node. The default host note is the last coupler specified. If the HN parameter is not specified, the terminal user must select a host before a connection can be made. If the AUTOCON parameter is specified, then the HN parameter must also be specified.

The value of this parameter does not change when the terminal class is changed from the console. The terminal user can change the value you specify.

IC PARAMETER

The IC parameter specifies whether or not the input mechanism of the device supports an ASCII DC3 code (X-OFF character) as a signal to stop input and an ASCII DC1 code (X-ON character) as a signal to resume input. CCP notifies the PAD to use these codes to control input flow when the device can support them.

This parameter allows you to configure a device so that input from an intelligent terminal such as a personal computer or from a cassette mechanism can be interrupted and restarted as needed without terminal user intervention. The PAD sometimes needs to suspend input on a line or circuit because the volume of network traffic has temporarily used all available storage space.

You should be careful that this parameter is appropriately defined for the device hardware that actually uses the line. These codes are used for many purposes by device manufacturers; receiving either code can have effects other than resuming or stopping input.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that CCP should not notify the PAD to use X-ON and X-OFF characters to control input.

YES   Indicates that CCP should notify the PAD to use X-ON and X-OFF characters to control input.

If you omit the IC parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	IC Value Used
H2000	NO
M33	NO
M40	NO
T4014	NO
2741	NO
713	NO
721	NO
X364	YES

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value you specify.

IN PARAMETER

The IN parameter identifies the input mechanism and the input message blocking of the device. (Terminal users can use a similar command that also defines whether input is in transparent mode or normalized mode; you cannot configure a device to begin accessing the network in transparent mode.)

As the input transmission mode, you can specify either:

Normalized line mode (one message or logical line per block).

Normalized block mode (one or more logical lines or messages collected into a block before it is transmitted. Cursor positioning at linefeed and end-of-line is not performed and output is not sent until end-of-block is reached.)

This parameter is optional; it is allowed for all terminal classes. Possible values are:

BK   Indicates keyboard input in block mode.

CCP   Indicates that CCP uses the default options appropriate for the terminal class; using this value is equivalent to omitting the parameter.

KB   Indicates keyboard input in line mode.

If you omit this parameter or specify the value CCP, the default value of KB is used (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

LI PARAMETER

The LI parameter specifies the number of idle characters to insert in the downline data after a linefeed code. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically move the carriage of the device to its next line for output.

This parameter is optional. If you declare this parameter, you must use the reserved word CCP or a value in the following range:

$$0 \leq li \leq 127$$

If you omit the LI parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	LI Value Used
H2000	3
M33	1
M40	3
T4014	0
713	0
721	0
X364	0

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

LK PARAMETER

The LK parameter specifies whether unsolicited messages from the NPU or host operator can appear at the terminal. This parameter is optional.

If you declare this parameter, you must specify one of the following values:

CCP   Indicates that CCP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that unsolicited messages should be delivered to the terminal as soon as received by CCP.

YES   Indicates that unsolicited messages should be discarded (locked out).

If you omit this parameter or specify the value CCP, the default value of NO is used (DT of CON only).

The application program or terminal user can also change the value you specify.

MCI PARAMETER

The MCI parameter specifies the delay after a carriage return is output. This delay is in 4-millisecond increments. This value is used by CCP to compute the number of idle characters output.

This parameter is optional. The MCI parameter has the following value range:

$$0 \leq mci \leq 250$$

Entering the terminal definition command CI to change the number of idle characters after a carriage return overrides the MCI value specified by either the application program or in the NDL. If the CI parameter and the MCI parameter are both specified in the NDL, whichever value is specified second overrides the first value. Entering the TC command to change the terminal class restores the default terminal class settings.

MLI PARAMETER

The MLI parameter specifies the delay after a line feed is output. This delay is in 4-millisecond increments. This value is used by CCP to compute the number of idle characters output.

This parameter is optional. The MLI parameter has the following value range:

$$0 \leq mli \leq 250$$

Entering the terminal definition command LI to change the number of idle characters after a line feed overrides the MLI value specified by either the application program or in the NDL. If the LI parameter and the MLI parameter are both specified in the NDL, whichever value is specified second overrides the first value. Entering the TC command to change the terminal class restores the default terminal class settings.

OC PARAMETER

The OC parameter specifies whether or not the output mechanism of the device sends an ASCII DC3 code (X-OFF character) as a signal for the PAD to interrupt output and an ASCII DC1 code (X-ON character) as a signal for the PAD to resume output. The PAD can accept these codes for control of output flow by devices that must periodically interrupt output to perform such functions as emptying buffers to offline storage devices. DC1 and DC3 codes used in this manner are discarded by the PAD.

You should only use this parameter when the device actually using the line requires it. The DC1 and DC3 codes can be sent for many purposes by terminal users; either code in input might be intended as data for the host application program.

This parameter is optional; it is allowed for all terminal classes except 2741. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that CCP should notify the PAD to ignore X-ON and X-OFF characters in input; output control by the device is unnecessary.

YES   Indicates that CCP should notify the PAD to recognize X-ON and X-OFF characters in input as output control by the device.

If you omit the OC parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	IC Value Used
H2000	NO
M33	NO
M40	NO
T4014	NO
2741	NO
713	NO
721	NO
X364	YES

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value you specify.

OP PARAMETER

The OP parameter identifies the output mechanism, page width processing requirement, and page length processing requirement of the device. CCP supports these possible output mechanisms:

A console screen

A hardcopy mechanism, such as a printer

CCP performs one of these processing actions when a finite page width is reached:

Inserts the codes appropriate to return the cursor or carriage to the beginning of the next physical line.

Inserts no codes, which performs no action.

If the device has a page width of 0, the second choice is always used. Refer to the description under the heading PW Parameter.

CCP performs one of these processing actions when a finite page length is reached:

Inserts the codes appropriate to cause the cursor or carriage to move to the next page (clear screen, formfeed, and so forth).

Inserts no codes, which performs no action.

If the device has a page length of 0, the second choice is always used. Refer to the PL Parameter description.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default options appropriate for the terminal class; using this value is equivalent to omitting the parameter.

DI   Indicates that the console display screen is used, no codes should be inserted when page width is reached (the device does its own line folding), and codes should be inserted when page length is reached.

PR   Indicates that a hardcopy printer is used, codes should be inserted when page width is reached (the device does not do its own line folding), and codes should be inserted when page length is reached.

If you omit this parameter or specify the value CCP, a default value is used. The default used depends on the TC parameter value declared or used; the default values are (DT of CON only):

TC Value	OP Value Used
M33	PR
713	DI
721	DI
X364	DI
H2000	DI
T4014	DI
M40	DI

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

PA PARAMETER

The PA parameter indicates the processing the X25 TIP should perform for the parity bit within each character byte of upline or downline data. This TIP processes the parity bit of all upline character codes in the following way:

Ignores the input byte parity bit after input. If the device is operating in normalized mode, sets the parity bit to zero and then forwards the byte upline to the application program. If the device is operating in transparent mode, the parity bit is set to zero for odd, even, and zero parity processing only; if parity is set to none or ignore, forwards the 8 bits unchanged to the application (for devices that send 8-bit bytes, this allows the application program to receive bit 7 as data). If the device is operating in transparent mode and parity is set to ignore, the parity bit is ignored when checking for the transparent mode delimiters.

CCP also processes the upper bit (bit 7) of all downline character code bytes in the following ways:

The parity bit in the output byte is set to zero, regardless of the setting of bit 7 in the downline byte. This is called zero parity processing.

The parity bit in the output byte is set to zero or one, as necessary to give the byte an odd number of set bits; the setting of bit 7 in the downline byte is ignored. This is called odd parity processing.

The parity bit in the output byte is set to zero or one, as necessary to give the byte an even number of set bits; the setting of bit 7 in the downline byte is ignored. This is called even parity processing.

The parity bit in the output byte is set to zero, regardless of the setting of bit 7 in the downline byte. If the terminal is operating in transparent mode, the parity bit in the output byte is unchanged from the setting of bit 7 in the downline byte (for devices that can receive 8-bit bytes, this allows the application program to use bit 7 as data). This is the method CCP uses for both no parity and ignore parity processing.

You can specify one of these input and output processing options to match the parity bit input processing performed by the device and the output processing expected by it. If the device will operate in transparent mode, you must specify either the no parity or the ignore parity option for an application program and the device to exchange hexadecimal codes with values between 7F and FF (using bit 7 as data).

This parameter is optional. If you specify the PA parameter, you must use one of the following values:

CCP   Indicates that CCP performs the parity processing appropriate as a default for the terminal class; using this value is equivalent to omitting the parameter.

E   Indicates that CCP should perform even parity processing for output bytes.

I   Indicates that CCP performs ignore parity processing for both input and output bytes.

N   Indicates that CCP should perform no parity processing for output bytes.

O   Indicates that CCP should perform odd parity processing for output bytes.

Z   Indicates that CCP should perform zero parity processing for output bytes.

If you omit this parameter or specify the value CCP, a default value of even parity processing is used.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

PG PARAMETER

The PG parameter specifies whether CCP should wait at each output page boundary for terminal user acknowledgment before it displays the next page of data. In certain situations, page waiting can occur other than at page boundaries; CCP produces a prompting message (OVER..) when this type of page waiting occurs. The user's response to page waiting is entry of a line, usually empty.

CCP views a new page as beginning at the start of each downline message. CCP calculates the length of a page from the current values of the page width and page length (see PW and PL Parameter descriptions). If the page width is infinite (PW=0), a page consists of one line less than the number of logical lines specified as the page length. If the page width is finite (PW is nonzero), a page consists of one line less than the number of physical lines specified by the page length; CCP calculates the number of physical lines by dividing each logical line into units less than or equal to the page width.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that CCP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that page waiting should not occur.

YES   Indicates that page waiting should occur.

When you specify PG=YES, you should also specify a nonzero value for the PL parameter. CCP cannot perform page waiting at the boundaries of infinitely long pages (PL=0).

If you omit the PG parameter or specify a value of CCP, the default value of NO is used (DT of CON only). The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

PL PARAMETER

The PL parameter specifies the number of physical lines per page of input or output for the device. If the device is configured for page waiting (PG parameter), any message containing more lines of output than the page length will be interrupted by CCP for a page waiting response from the terminal user. The interruption occurs after line pl - 1 is output. If the device is defined with a printer as its output device (OP value of PR), CCP inserts formfeed codes at page length boundaries.

This parameter is optional. If you specify the PL parameter, you must use one of the following values:

CCP

     Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

     Indicates an infinite page length. An infinite page length means that no page waiting occurs and no formfeed codes are inserted.

$8 \leq pl \leq 255$

Indicates the number of physical lines per page.

If you omit the PL parameter or specify the value CCP, a default value is used (DT of CON only). The default value used depends on the TC parameter value used for the terminal. The default values are:

TC Value	PL Value Used
M33	0
713	24
721	30
M40	24
H2000	27
X364	24
T4014	35

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

PRI PARAMETER

The PRI parameter is a stand-alone keyword. This keyword indicates whether data to or from the device is to have traffic priority over that to or from other devices in the network. Data to or from a device that does not have traffic priority is suspended to relieve network congestion before data to or from terminals with traffic priority is suspended. Interactive terminals should usually be given traffic priority over batch devices and application-to-application connections.

This parameter is optional. If you specify the PRI parameter, you must use one of the following values:

NO   Indicates that the circuit should not have data traffic priority.

YES   Indicates that the circuit should have data traffic priority.

If you declare this parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

PW PARAMETER

The PW parameter defines the number of characters per physical line of input or output for this device. This physical line length is also called the page width.

For console devices (DT of CON) with a printer defined as the output mechanism (OP of PR), output lines longer than pw characters are divided into lines of pw or fewer characters each. For console devices with a display defined as the output mechanism (OP of DI), output lines longer than pw characters might be divided into lines of pw or fewer characters each by the host application program; if the PW value for a device is inappropriate, loss of visual fidelity can occur (the application program might divide the data into lines that are too short or too long for the screen's capacity, instead of allowing the terminal to wrap lines when needed).

This parameter is optional. If you specify the PW parameter, you must use one of the following values:

CCP

Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

Indicates that the device has an infinite page width. An infinite page width means that physical line length has no effect on output line formatting.

$20 \leq pw \leq 255$

Indicates that the device can support physical lines no longer than the indicated decimal value.

If you omit this parameter or specify the value CCP, a default value is used. The default value depends on the terminal class. Default values for console devices (DT of CON) are:

TC Value	PW Value Used
M33	72
713	80
721	80
X364	80
M40	80
H2000	74
T4014	74

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or terminal user can also change the value you specify.

P90 THROUGH P99 PARAMETERS

These ten parameters indicate the hexadecimal field value to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this device. The released version of the CDC-defined X25 subTIP does not use any of these ten field number/field value pairs.

These parameters are optional. If you specify a value, it must be within the following range:

$0 \leq fv9i \leq FF$

If you omit these parameters, there are no default values.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input messages and if you choose a UBL value smaller than this number, the application program becomes deadlocked (it waits for blocks of data that CCP is forced to discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following value range:

$$1 \leq ubl \leq 31$$

The NPU queues all upline blocks. Large UBL values rapidly use up NPU memory and cause slower data transfers.

If you omit the UBL parameter, a default value is used (DT of CON only). The default is 7. The application program can change the value you specify.

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline block can contain. CCP divides each message from the terminal into blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data the NPU and the host each queue.

You should choose a value that allows most of the logical lines entered by the device to fit into a single block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional. The UBZ parameter has the following value range for console devices (DT of CON):

$$0 \leq ubz \leq 2000$$

Site-defined devices can use the following range:

$$0 \leq ubz \leq 2043$$

If you specify 0, CCP sends an upline block whenever it receives 100 characters or it detects a linefeed character.

The value you declare should be chosen together with the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes, as follows:

Value Supplied	Value Used in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

The NDL processor issues an informative message if it adjusts the UBZ value that you specify.

If you omit the UBZ parameter, a default value is used (DT of CON only). The default is 100. This default optimizes use of the default value for the UBL parameter.

For the Message Control System (MCS), the upline block size must be set to 0.

The application program can change any value you specify. The terminal user can change the value within the following range:

$$0 \leq ubz \leq 200$$

XLC PARAMETER

The XLC parameter indicates the maximum number of characters that can be input in each message of multiple-message transparent mode input from this device. After the terminal user or the application program changes the device's input mode to transparent from normalized, the following occurs. When the device transmits the given number of characters, a message block is forwarded to the application and the device remains in transparent mode.

This parameter is optional. You cannot use the XLC parameter if you use the DLC, DLTO, or DLX parameters. If specified, XLX or XLTO must also be specified. You can use one of the following values:

CCP

> Indicates that CCP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$$1 \leq xlc \leq 4095$$

> Indicates the maximum decimal number of characters that the terminal can transmit as one transparent mode message.

If you omit this parameter or specify the value CCP, no default exists.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value you specify.

XLTO PARAMETER

The XLTO parameter specifies whether an end-of-packet ends multiple message transparent mode input. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP forwards a message block to the application and changes back from transparent mode to normalized mode when the device transmits the end-of-packet sequence.

This parameter is optional. You cannot use the XLTO parameter if you use the DLC, DLTO, or DLX parameters. You can use one of the following values:

CCP   Indicates that CCP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that a timeout does not end transparent mode input.

YES   Indicates that transparent mode input ends when a timeout occurs.

If you omit this parameter or specify the value CCP, no default exists.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can also change the value you specify.

XLX PARAMETER

The XLX parameter indicates the hexadecimal value of the character code that should cause upline transmission of a message in multiple-message transparent mode. If this character code is also used as the value for the XLY parameter and is input twice in succession, the device will be put back in normalized mode as determined by the IN parameter.

This parameter is optional. You cannot use the XLX parameter if you use the DLC, DLTO, or DLX parameters.

The value that is valid for a given character depends on the character code set used by the terminal. You can use one of the following values:

CCP

> Indicates that CCP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \leq xlx \leq FF$

> Indicates the hexadecimal code that ends each message within multiple-message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F are not seen by CCP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, no default exists.

The application program or the terminal user can change any value you specify.

XLY PARAMETER

The XLY parameter indicates the hexadecimal value of the character code that ends multiple-message transparent mode operation when entered following the XLX character code. After the terminal user or the application program changes the device's input mode to transparent from normalized, CCP changes back from transparent mode to normalized mode when the device transmits this character immediately after the character defined for the XLX parameter. If this character is also used for the XLX parameter, the code does not change the device's input mode unless it is input twice in succession.

You should select transparent mode input delimiters with care. The character code you declare as the XLY value must be the code of a character that the physical terminal can input. If the delimiter you declare is a character that cannot be input (either because of terminal hardware limitations or because of the PA parameter value declared for the terminal), then the Terminal Interface Program cannot terminate transparent mode input and the terminal will be trapped in that mode of operation once it has begun.

This parameter is optional. You cannot use the XLY parameter if you use the DLC, DLTO, or DLX parameters. If you use this parameter, you must use the XLX parameter.

The value that is valid for a given character depends on the character code set used by the terminal. You can use one of the following values:

CCP

> Indicates that CCP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \leq xly \leq FF$

> Indicates the hexadecimal code that ends multiple-message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F are not seen by CCP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, and XLX is also specified, the default is the code specified for XLX. Otherwise, no default exists.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

APPLICATION-TO-APPLICATION CONNECTION DEFINITIONS

Each X.25 application-to-application connection definition consists of a TERMINAL definition and a DEVICE definition and represents one or more switched virtual circuits. Each virtual circuit can service one application connection.

Switched virtual circuits are comparable to terminals using dialup lines in that the characteristics of the device using the line can change from use to use of the line.

You specify only one application-to-application connection definition for all of the application connections that can use the line. As each switched virtual circuit becomes active, CCP creates a separate device definition for it, creating an element name for each definition. This device name consists of the one to five characters you declare, plus an ordinal consisting of the switched virtual circuit number.

TERMINAL DEFINITIONS

You must provide one TERMINAL or TERMDEV statement for the application connections that can access the link. Figures 5-5 and 5-6 present the format of these statements for application-to-application connections on X.25 lines.

```
TERMINAL[,STIP=stiptyp,CSET=charset,]

[NCIR=numcir,NEN=encir].

Parameters are described in the text.
```

Figure 5-5. TERMINAL Statement Format
for Communication Lines of TIPTYPE=X25

The LINE statement must specify the total number of circuits available from the packet-switching network via the line (NSVC parameter). You can declare only one TERMINAL and DEVICE statement or one TERMDEV statement for each subTIPtype (STIP parameter) on the line. You can use the TERMINAL or TERMDEV statement NCIR parameter to define more than one circuit with that statement. The NCIR parameter determines the highest ordinal number used to generate a unique element name for each device using the circuit.

The following parameters provide the terminal definition for the line:

    CSET
    NCIR
    NEN
    STIP

The STIP parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

STIP Parameter

The STIP parameter specifies the packet protocol required to support the terminal definition. This parameter is required for application-to-application connections.

When you declare a value for STIP, it must be one of the following reserved words:

    XAA   Identifies   the   application-to-
           application X.25 subTIP.

CSET Parameter

The CSET parameter specifies the code and character set of the terminal. The CSET parameter is optional.

When you declare a value for CSET, you must use one of the following words:

    ASCII   ASCII code and character set

    CSET15   Site-defined code and character set,
             identified within the network soft-
             ware as character set number 15

If you omit CSET, the default value of ASCII is used.

NCIR Parameter

The NCIR parameter specifies the decimal number of virtual circuits of the same subTIPtype you are defining. This parameter is optional.

If you specify this parameter, the value you use must be less than or equal to the NSVC value declared or used on the corresponding LINE statement. The NCIR parameter has the following range of values:

$$1 \leq numcir \leq 255$$

If you omit this parameter, the default value of 1 is used.

NEN Parameter

The NEN parameter specifies the decimal number of virtual circuits of the same subTIPtype that are intially enabled. This parameter is optional.

If you specify this parameter, the value you use must be less than or equal to the NCIR value declared or used on the same TERMINAL or TERMDEV statement. The NEN parameter has the following range of values:

$$1 \leq encir \leq 255$$

If you omit this parameter, the default value of 1 is used.

DEVICE DEFINITIONS

You must provide one TERMDEV or DEVICE statement for all switched virtual circuit application-to-application connections that can access the communication line through the terminal. Each terminal statement can have only one DEVICE statement. Figures 5-6 and 5-7 present the formats of the TERMDEV and DEVICE statements for terminals on X.25

The following parameters provide the definition for a CDC-defined X.25 terminal device:

DT

DT Parameter

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

When you declare the DT parameter, the following values are valid:

AP   Identifies an interhost application-to-application connection.

DT12   Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

If you omit the DT parameter, AP is used for STIP=XAA.

BUFFERING OF DATA

It is desirable to maintain double buffering of data for each connection in the network. By using double buffering data will always be available to be sent over the communications line. Setting the ABL parameter to 2 and the DBL parameter to 1 will generally result in double buffering.

```
device:   TERMDEV[,STIP=stiptyp,CSET=charset,NCIR=numcir,NEN=encir,DT=devtyp].


device       The element name of the connection being defined.  This name can be one through five
             characters long.  The first character must be a letter; the other characters can be
             letters or digits.  This name concatenated with the virtual circuit number is used by the
             host or NPU operator to monitor and control the connection.  This name is required; there
             is no default value.

All other parameters are described in the text.
```

Figure 5-6. TERMDEV Statement Format for Application Programs Using Communication Lines of TIPTYPE=X25

```
device:   DEVICE[,DT=devtyp].


device       The element name of the connection being defined.  This name can be one through five
             characters long.  The first character must be a letter; the other characters can be
             letters or digits.  This name concatenated with the virtual circuit number is used by the
             host or NPU operator to monitor and control the connection.  This name is required; there
             is no default value.

All other parameters are described in the text.
```

Figure 5-7. DEVICE Statement Format for Application Programs Using Communication Lines of TIPTYPE=X25

CCP buffer usage is optimized when downline block size follows the following relationship:

$$DBZ = 113 + n*118$$

where n is an integer and n + 1 is the number of buffers required in CCP to hold the block. This relationship is of most use for application programs that can control the size of the downline blocks they send to the network. Since it is necessary for a block to contain an integral number of logical lines, the lower bound of DBZ should not be smaller than the largest allowable PW for a device.

Default values for the ABL, DBL, and DBZ parameters are listed in table 5-2.

TABLE 5-2. ABL, DBL, AND DBZ DEFAULTS

Line Speed	ABL	DBL	DBZ
110	1	1	230
300	1	1	230
600	2	1	230
1200	2	1	230
2400	2	1	230
4800	2	1	460
9600	2	1	940

The parameters required and the values allowed on the GROUP, LINE, TERMINAL, TERMDEV, and DEVICE statements depend upon the line protocol used. This section describes the form of a statement applicable to synchronous protocol lines.

Synchronous communication lines are those using the line types defined in table 6-1. Such lines are serviced by the CDC-supplied MODE4, HASP, BSC, or 3270 TIPs. If you use the site-defined line type, one of these TIPs is presumed to be modified by your site (if necessary) to support either a standard communication line adapter or a specially built communication line adapter.

LINE DEFINITION

You configure terminals on a synchronous communication line by using the following statements:

One LINE or GROUP statement that defines the line.

One TERMINAL or TERMDEV statement for each potential terminal on the line (more than one mode 4C protocol terminal can access the line at a given time; multiple-drop synchronous lines of this type are supported).

One DEVICE statement for each device that accesses the line through a terminal; if you use a TERMDEV statement, you cannot use DEVICE statements for the same terminal.

LINE STATEMENT PARAMETERS

Each LINE statement defines one communication line between a terminal and the NPU. There must be one LINE statement for each CLA port on the NPU that supports terminal access.

Figure 6-1 shows the format of the LINE statement and the valid parameter values for CDC-defined synchronous lines. Table 6-1 shows permitted line type values for the LTYPE parameter.

TABLE 6-1. SYNCHRONOUS LINE TYPE DEFINITIONS

LTYPE Value	Transmission Mode	Transmission Operation	Circuit Type	Modem Type†	CLA Type	Maximum Speed, Bits per Second	Carrier Type
S1	Synchronous	Half-duplex	Switched (Dialup)	RS232C, Bell 201A/208B compatible	2560-1	4800	Controlled
S2	Synchronous	Full-duplex but operating half-duplex	Dedicated (Hard-wired)	RS232C, Bell 201B/208A compatible	2560-1	19200	Controlled
				Bell 301/303 compatible	2560-2	50000 (HASP only)	
				V.35 Standard compatible	2560-3	56000 (HASP only)	
S3	Synchronous	Full-duplex	Dedicated (Hard-wired)	RS232C, Bell 201B/208A compatible	2560-1	19200	Constant
				Bell 301/303 compatible	2560-2	50000 (HASP only)	
				V.35 Standard compatible	2560-3	56000 (HASP only)	
S4 (For site-defined use)	Synchronous	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned

†Modem types supported by each type of CLA may differ.

```
Line:    LINE,PORT=port,LTYPE=ltype[,TIPTYPE=tiptype,AUTO[=yn1],DI[=yn2],AL=acclev,IMDISC[=yn3],]

    [RC[=yn4],P90=fv90,...,P99=fv99].
```

Line   The element name to be assigned to the communication line being defined. This name is
                  required; there is no default value.

port   The hexadecimal number (1 $\leq$ port $\leq$ FE) of the port to which this communication line connects
                  on the NPU currently being defined. Within an NDL program, the port number is independent
                  of the number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE
                  (254 decimal). However, we strongly recommend that you assign port numbers consecutively,
                  starting with 1. All values declared for PORT parameters must be unique within the current
                  network definition of each NPU. The value specified for port cannot the same as the
                  number declared for a port in any other LINE or TRUNK statement (or within a GROUP statement
                  expansion) for this NPU, and cannot be lower than the highest port number used by a trunk.
                  This number is required; there is no default value.

ltype   A reserved word value that identifies the type of communication line adapter/modem/circuit
                  combination that is used on this line. This word must be supplied; there is no default
                  value. The legal words for this value declaration are described in table 6-1.

tiptype   A reserved word value that identifies the type of CCP Terminal Interface Program protocol
                  required for the terminals on this line. This word is optional for lines being configured
                  for automatic recognition of a synchronous protocol terminal; otherwise it must be supplied
                  (there is no default value). The legal words for this value declaration are:

                      BSC   2780/3780 binary synchronous protocol TIP is required.
                      HASP   HASP multileaving workstation binary synchronous protocol TIP is required.
                      MODE4   Control Data mode 4 protocol TIP is required.
                      3270   3270 binary synchronous protocol TIP is required.

yn1   An optional reserved word value (YES or NO) for an optional stand-alone keyword. This
                  parameter specifies whether this communication line is configured for automatic recognition
                  of protocol and subTIPtype by CCP whenever a terminal is connected to the line. When AUTO
                  or AUTO=YES is specified, the line is configured for automatic recognition. When automatic
                  recognition is performed, CCP attempts to determine as many addressing parameters as
                  possible at the time the line becomes active; recognized parameters are compared against
                  parameters specified in the network definition file to identify the terminal accessing the
                  line. If AUTO is omitted or AUTO=NO is specified, CCP is not allowed to perform automatic
                  recognition for the line; the line has a fixed configuration, and each TERMINAL or TERMDEV
                  statement associated with this line must explicitly declare all addressing and protocol
                  parameters legal for that terminal. AUTO cannot be specified for lines with TIPTYPE=3270.

yn2   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this
                  parameter specifies the status assigned to this communication line at network initiation.
                  When DI or DI=YES is specified, the line is disabled at network initiation and cannot be
                  used until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is
                  given an initial status of enabled unless the NOP or HOP specifies otherwise.

acclev   The decimal access level limit (0 $\leq$ acclev $\leq$ 7) for the line. The lowest access level is 0
                  (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level
                  limit is associated with the line (equivalent to specifying AL=0).

yn3   An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect
                  feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on
                  the line is disconnected immediately after the terminal has logged out (disconnected) from
                  the host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is
                  disconnected only after a two minute timer expires.

yn4   An optional reserved word value (YES or NO) which specifies whether the reconfiguration
                  indicator is enabled for this line. When RC or RC=YES is specified, the terminal
                  characteristics are reset to their original NDL values (or to a default if no NDL values
                  were specified) when the terminal disconnects from a host. When RC is omitted or RC=NO is
                  specified, the reconfiguration indicator is not enabled for this line.

fv9i   The hexadecimal field value (0 $\leq$ fv9i $\leq$ FF) to use for the corresponding field number within
                  the Terminal Interface Program configuration information transmitted for this line. The
                  released version of the CDC-written MODE4, HASP, BSC, and 3270 TIPs do not use any of these
                  ten field number/field value pairs.

Figure 6-1. Synchronous LINE Statement Format

You use the AUTO parameter to define a communication line as having a fixed or an automatic recognition configuration. You can use either of these configuration types for a switchable (dialup) line or for a dedicated (hardwired) line.

Auto-recognition of 3270 terminals is not performed. For this reason, dial-up 3270 terminals must dial into synchronous ports that support only 3270s. The LINE definition for a 3270 terminal must include all cluster and terminal addresses that can connect to the port.

If you define a fixed-configuration line by omitting AUTO or specifying AUTO=NO, you must know and specify certain characteristics of the terminals and devices that will use it. Only terminals and devices with the specified characteristics can use that line.

The characteristics of a fixed-configuration line are:

CCP Terminal Interface Program type (LINE statement TIPTYPE parameter)

SubTIPtype (TERMINAL or TERMDEV statement STIP parameter)

Cluster addresses (TERMINAL or TERMDEV statement CA parameter; required for mode 4 and 3270 protocol lines only). Hardwired fixed-configuration lines can support multiple mode 4 terminals

Code and character set (TERMINAL or TERMDEV statement CSET parameter)

Terminal address (TERMDEV or DEVICE statement TA parameter; required for mode 4, BSC, and 3270 protocol lines only)

Device stream number (DEVICE statement STREAM parameter; required for HASP protocol lines only)

Device type (TERMDEV or DEVICE statement DT parameter)

Number of devices (implied from number of TERMDEV or DEVICE statements)

If you define an automatic recognition line by specifying AUTO or AUTO=YES, CCP determines these characteristics when a terminal becomes active on the line. This allows more flexible access to the network.

For mode 4 protocol terminals, CCP determines the automatic recognition information from the first codes it receives when the terminal becomes active. For HASP terminals, CCP requires one of the following:

The protocol signon block must contain a /*CONFIG card image with the same configuration ordinal and device number information as the definition you create for that terminal.

A /*CONFIG card image must be the first input from the terminal. This card must contain the same configuration ordinal and device number information as the definition you create for that terminal.

For bisynchronous terminals, CCP requires that a /*CONFIG card image be the first input from the terminal. This card must contain the same configuration ordinal and device number information as the definition you create for that terminal.

A terminal on an automatic recognition line is not completely configured until it becomes active. At that time, the Communications Supervisor (CS) compares the determined characteristics against the characteristics you have defined for each terminal configured on the line. CS uses the first terminal definition that matches to finish configuring the terminal. CCP then services the terminal devices according to the finished configuration.

Only a terminal that completely matches all declared values can use the line. It must operate with the characteristics you declare for any required or optional parameters in your terminal definition, or the terminal user must change those characteristics to match. If you declare AUTOREC instead of values for all of the automatically recognized parameters, then any terminal will match the terminal definition.

If you declare values instead of AUTOREC for some of the automatically recognized parameters, then any terminal that successfully accesses the line and has characteristics matching the declared values will match your terminal definition. If you declare values instead of AUTOREC for all of the automatically recognized parameters, then only a terminal with automatically recognized characteristics that are the same as your values will match the terminal definition.

You can specify more than one terminal definition for terminals of each protocol (each TIP type) on an automatic recognition line. Each definition should vary from the others in one of the automatically recognized characteristics; unless differences exist, CS will not use any definition other than the first one. If you specify terminals of more than one protocol, only terminals of one TIP type can access the line at the same time.

Using automatic recognition increases the number of terminals that potentially can use the line. You can configure a switchable line for automatic recognition with more logical terminals than the physical terminals that simultaneously can access it. However, only one physical HASP, mode 4A protocol, or bisynchronous terminal can access the line at a given time.

You can configure a mode 4 line for automatic recognition without specifying cluster addresses on TERMINAL statements or terminal addresses on DEVICE statements. Only one physical mode 4C protocol terminal with a given cluster address can access the line at a given time, and only one device with a given terminal address can be configured at a given time.

For example, the statement

LN3F: LINE PORT=10,LTYPE=S1,AUTO.

defines a switchable (dialup), synchronous line (on NPUF in SVLnet, figure 2-1 of section 2) for automatic recognition of any synchronous protocol terminals. The line is identified as LN3F, connects to the NPU at port 10, and is enabled by default at network initiation.

If the automatic recognition form of the statement

    LN3F: LINE PORT=10,LTYPE=S1,TIPTYPE=MODE4,AUTO.

were used instead, only mode 4 terminals could use the line.

As an example of a fixed-configuration line, the statement

    LN2E: LINE PORT=10,LTYPE=S3,TIPTYPE=HASP.

defines a hardwired (dedicated), enabled, synchronous line for multileaving workstation terminals, accessing the NPU (NPUE in SVLnet) through port 10.

All of the terminals capable of accessing this switchable line would have to be completely and explicitly configured on their TERMINAL and DEVICE statements. However, the characteristics of the terminal can be automatically recognized by the MODE4 TIP.

GROUP STATEMENT PARAMETERS

You can use the GROUP statement in place of the LINE statement when you want to repeat a line definition a specified number of times.

NDLP creates identical line, terminal, and device definitions (except for port number, line name, and device name) the number of times specified by the NI parameter. NDLP generates unique port numbers by incrementing the previously defined port number by one. NDLP generates unique line and device element names by adding the LINE's port number in two-digit hexadecimal form to the one- to five-character specified root element name.

The format of the GROUP statement is shown in figure 6-2. Table 6-1 shows permitted line type values for the LTYPE parameter.

Using the GROUP statement can reduce the number of definitions you must provide in your NDL program without reducing the number of definitions provided in the network configuration file. An example of GROUP statement use is shown in figure 6-3.

You use the AUTO parameter to define a communication line as having a fixed or an automatic recognition configuration. You can use either of these configuration types for a dialup line or for a dedicated (hardwired) line.

Auto-recognition of 3270 terminals is not performed. For this reason, dial-up 3270 terminals must dial into synchronous ports that support only 3270s. The LINE definition for a 3270 terminal must include all cluster and terminal addresses that can connect to the port.

If you define a fixed-configuration line by omitting AUTO or specifying AUTO=NO, you must know and specify certain characteristics of the terminals and devices that will use it. Only terminals and devices with the specified characteristics can use that line.

The characteristics of a fixed-configuration line depend on the communication protocol used by the line. These characteristics are:

    CCP Terminal Interface Program type (GROUP statement TIPTYPE parameter)

    SubTIPtype (TERMINAL or TERMDEV statement STIP parameter)

```
group:  GROUP,PORT=port,LTYPE=ltype[,TIPTYPE=tiptype,AUTO[=yn1],DI[=yn2],AL=acclev,IMDISC[=yn3],]

        [RC[=yn4],P90=fv90,...,P99=fv99,NI=iter].
```

group	The root name to use for generating the element names assigned to the communication lines and devices being defined. This name, which cannot be longer than five characters, must result in element names that are unique within the network division currently being described. This name is required; there is no default value.
port	The hexadecimal number (1 ≤ port ≤ FE) of the port to which the first of these communication lines connects on the NPU currently being defined. Within an NDL program, the port number is independent of the number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE (254 decimal). However, we strongly recommend that you assign port numbers consecutively, starting with 1. All values declared for PORT parameters must be unique within the current network definition of each NPU. The value specified for port cannot be the same as the number declared for a port in any LINE or TRUNK statement (or within any other GROUP statement expansion) for this NPU, and cannot be lower than the highest port number used by a trunk. This number is required; there is no default value.
ltype	A reserved word value that identifies the type of communication line adapter/modem/circuit combination that is used on this line. This word must be supplied; there is no default value. The legal words for this value declaration are described in table 6-1.

tiptype	A reserved word value that identifies the type of CCP Terminal Interface Program protocol required for the terminals on this line. This word is optional for lines being configured for automatic recognition of a synchronous protocol terminal; otherwise, it must be supplied (there is no default value). The legal words for this value declaration are:
	
	  BSC Binary synchronous protocol TIP is required.
	  HASP HASP multileaving workstation binary synchronous protocol TIP is required.
	  MODE4 Control Data mode 4 protocol TIP is required.
	  3270 3270 protocol TIP is required.

Figure 6-2. Synchronous GROUP Statement Format (Sheet 1 of 2)

yn1	An optional reserved word value (YES or NO) for an optional stand-alone keyword. This parameter specifies whether this communication line is configured for automatic recognition of protocol and subTIPtype by CCP whenever a terminal is connected to the line. When AUTO or AUTO=YES is specified, the line is configured for automatic recognition. When automatic recognition is performed, CCP attempts to determine as many addressing parameters as possible at the time the line becomes active; recognized parameters are compared against parameters specified in the network definition file to identify the terminal accessing the line. If AUTO is omitted or AUTO=NO is specified, CCP is not allowed to perform automatic recognition for the line; the line has a fixed configuration, and each TERMINAL or TERMDEV statement associated with this line must explicitly declare all addressing and protocol parameters legal for that terminal. AUTO cannot be specified for lines with TIPTYPE=3270.
yn2	An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter specifies the status assigned to this communication line at network initiation. When DI or DI=YES is specified, the line is disabled at network initiation and cannot be used until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is given an initial status of enabled unless the NOP or HOP specifies otherwise.
acclev	The decimal access level limit (0 $\leq$ acclev $\leq$ 7) for the line. The lowest access level is 0 (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level limit is associated with the line (equivalent to specifying AL=0).
yn3	An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on the line is disconnected immediately after the terminal has logged out (disconnected) from the host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is disconnected only after a two minute timer expires.
yn4	An optional reserved word value (YES or NO) which specifies whether the reconfiguration indicator is enabled for this line. When RC or RC=YES is specified, the terminal characteristics are reset to their original NDL values (or to a default if no NDL values were specified) when the terminal disconnects from a host. When RC is omitted or RC=NO is specified, the reconfiguration indicator is not enabled for this line.
fv9i	The hexadecimal field value (0 $\leq$ fv9i $\leq$ FF) to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this line. The released version of the CDC-written MODE4, HASP, BSC, and 3270 TIPs do not use any of these ten field number/field value pairs.
iter	The decimal number (1 $\leq$ iter $\leq$ 254) of iterations of this line definition and of all TERMINAL, TERMDEV, or DEVICE statements following this GROUP statement. This parameter is optional; the default is 1.

Figure 6-2. Synchronous GROUP Statement Format (Sheet 2 of 2)

Cluster addresses (TERMINAL or TERMDEV statement CA parameter; required for mode 4 and 3270 protocol lines only). Hardwired fixed-configuration lines can support multiple mode 4 terminals.

Code and character set (TERMINAL or TERMDEV statement CSET parameter)

Terminal address (TERMDEV or DEVICE statement TA parameter; required for mode 4, BSC, and 3270 protocol lines only)

Device stream number (DEVICE statement STREAM parameter; required for HASP protocol lines only)

Device type (TERMDEV or DEVICE statement DT parameter)

Number of devices (implied from number of TERMDEV or DEVICE statements)

If you define an automatic recognition line by specifying AUTO or AUTO=YES, CCP determines these characteristics when a terminal becomes active on the line. This allows more flexible access to the network.

For mode 4 protocol terminals, CCP determines the automatic recognition information from the first codes it receives when the terminal becomes active. For HASP terminals, CCP requires one of the following:

The protocol signon block must contain a /*CONFIG card image with the same configuration ordinal and device number information as the definition you create for that terminal.

A /*CONFIG card image must be the first input from the terminal. This card must contain the same configuration ordinal and device number information as the definition you create for that terminal.

For bisynchronous terminals, CCP requires that a /*CONFIG card image be the first input from the terminal. This card must contain the same configuration ordinal and device number information as the definition you create for that terminal.

A terminal on an automatic recognition line is not completely configured until it becomes active. At that time, the Communications Supervisor (CS) compares the determined characteristics against the

```
The following set of statements:

    LN3F:  GROUP,PORT=0A,LTYPE=S2,

        TIPTYPE=MODE4,NI=2.

        TERMINAL,STIP=M4A,TC=200UT,CA=70.

            DEVD:  DEVICE,DT=CON,TA=60.

            DEVL:  DEVICE,DT=LP,TA=60.

            DEVC:  DEVICE,DT=CR,TA=60.

is interpreted by the NDL processor as if it had
been written:

    LN3F0A:  LINE,PORT=0A,LTYPE=S2,

        TIPTYPE=MODE4.

        TERMINAL,STIP=M4A,TC=200UT,CA=70.

            DEVD0A:  DEVICE,DT=CON,TA=60.

            DEVL0A:  DEVICE,DT=LP,TA=60.

            DEVC0A:  DEVICE,DT=CR,TA=60.

    LN3F0B:  LINE,PORT=0B,LTYPE=S2,

        TIPTYPE=MODE4.

        TERMINAL,STIP=M4A,TC=200UT,CA=70.

            DEVD0B:  DEVICE,DT=CON,TA=60.

            DEVL0B:  DEVICE,DT=LP,TA=60.

            DEVC0B:  DEVICE,DT=CR,TA=60.
```

Figure 6-3.   Synchronous GROUP Statement
Expansion

characteristics you have defined for each terminal
configured on the line. CS uses the first terminal
definition that matches to finish configuring the
terminal. CCP then services the terminal devices
according to the finished configuration.

Only a terminal that completely matches all
declared values can use the line. It must oper-
ate with the characteristics you declare for any
required or optional parameters in your terminal
definition, or the terminal user must change the
characteristics to match. If you declare AUTOREC
instead of values for all of the automatically
recognized parameters, then any terminal will match
the terminal definition.

If you declare values instead of AUTOREC for some
of the automatically recognized parameters, then
any terminal that successfully accesses the line
and has characteristics matching the declared
values will match your terminal definition. If you
declare values instead of AUTOREC for all of the
automatically recognized parameters, then only a
terminal with automatically recognized character-
istics that are the same as your values will match
the terminal definition.

You can specify more than one terminal definition
for terminals of each protocol (each TIP type) on
an automatic recognition line. Each definition

should vary from the others in one of the automat-
ically recognized characteristics; unless differ-
ences exist, CS will not use any definition other
than the first one. If you specify terminals of
more than one protocol, only terminals of one TIP
type can access the line at the same time.

Using automatic recognition increases the number of
terminals that potentially can use the line. You
can configure a switchable line for automatic
recognition with more logical terminals than the
physical terminals that simultaneously can access
it. However, only one physical HASP, mode 4A
protocol, or bisynchronous terminal can access the
line at a given time.

You can configure a mode 4 line for automatic
recognition without specifying cluster addresses on
TERMINAL statements or terminal addresses on DEVICE
statements. Only one physical mode 4C protocol
terminal with a given cluster address can access
the line at a given time, and only one device with
a given terminal address can be configured at a
given time.

For example, the statement

    LN3F: GROUP,PORT=0A,LTYPE=S1,AUTO,NI=2.

defines two switchable (dialup), synchronous lines
(not shown in figure 2-1 of section 2) for automatic
recognition of synchronous protocol terminals. The
lines are identified as LN3F0A and LN3F0B, connect
to the NPU at ports 0A and 0B, and are enabled by
default at network initiation.

If the automatic recognition form of the statement

    LN3F: GROUP,PORT=0A,LTYPE=S1,TIPTYPE=MODE4,
           AUTO,NI=2.

were used instead, only mode 4 terminals could use
the lines.

As an example of a fixed-configuration line pair,
the statement

    LN2E: GROUP,PORT=0A,LTYPE=S3,TIPTYPE=HASP,NI=2.

defines two hardwired (dedicated), enabled,
synchronous lines for multileaving workstation ter-
minals, accessing the NPU through ports 0A and 0B.

All of the terminals capable of accessing these
switchable lines would have to be completely and
explicitly configured on the TERMINAL and DEVICE
statements following the GROUP statement. However,
the characteristics of the terminal can be auto-
matically recognized by the MODE4 TIP.

MODE 4 TERMINAL DEFINITIONS

On a mode 4 communication line, a terminal defini-
tion is different from a device definition. A mode
4 terminal consists of one or more devices; ter-
minal and device are not synonymous. Within the
network, a mode 4 terminal is a cluster controller;
this is synonymous with a mode 4A site (equipment
controller) or with a mode 4C station.

If you defined the communication line as a fixed
definition line, you can declare up to 16 TERMINAL
or TERMDEV statements for the line (if it is a
multiple-drop mode 4 line). If you defined the

line as an automatic recognition line, you can declare up to 255 TERMINAL or TERMDEV statements for the line. Only one can be connected at a time.

You must provide one TERMINAL or TERMDEV statement for each terminal that can access the communication line simultaneously. Figures 6-4 and 6-5 present the formats of these statements for terminals on mode 4 protocol synchronous lines.

TERMINAL[,STIP=stiptyp,TC=trmclas,]

[CSET=charset,CA=clsadr,]

[RIC=yn1,EOF[=yn2]].

Parameters are described in the text.

Figure 6-4. TERMINAL Statement Format
for Communication Lines of TIPTYPE=MODE4

The following TERMINAL and TERMDEV parameters provide the terminal definition for the line:

    CA
    CSET
    EOF
    RIC
    STIP
    TC

The STIP and TC parameters are described first because they are related and affect the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

STIP PARAMETER

The STIP parameter specifies the subcategory of the TIP type (the subTIPtype) required to support the terminal definition. This parameter is optional if the TC parameter specifies a CDC-defined terminal class. If STIP is omitted, the TC parameter must be specified and cannot be CCP or TC28 through TC31.

When you declare a value for STIP, it must be one of the following reserved words:

AUTOREC   Indicates that CCP should use the subTIPtype appropriate for the terminal class (valid only if TC is specified); using this value is equivalent to omitting the parameter.

M4A   Identifies a synchronous terminal that uses the mode 4A protocol variant.

M4C   Identifies a synchronous terminal that uses the mode 4C protocol variant.

If you use the AUTOREC value or omit the STIP parameter, the default is selected according to the terminal class declared. The defaults are:

TC Value	STIP Value Used
200UT	M4A
734	M4A
711	M4C
714	M4C
714X	M4C

For console devices:

device: TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,CA=clsadr,RIC=yn1,DT=CON,]

   [TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

   [AUTOCON[=yn2],PRI[=yn3],DI[=yn4],HN=node,HD[=yn5],LK=yn6,B1=b1,ELO=elo,B2=b2,CN=cn,]

   [CT=ct,PG=pg,PL=pl,PW=pw,EOF[=yn7],P90=fv90,...,P99=fv99].

For site-defined devices:

device: TERMDEV,[STIP=stiptyp,TC=trmclas,CSET=charset,CA=clsadr,RIC=yn1,]DT=DT12[,]

   [SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

   [XBZ=xmitsiz,DO=devord,AUTOCON[=yn2],PRI[=yn3],DI[=yn4],HN=node,HD[=yn5],LK=yn6,]

   [B1=b1,B2=b2,CN=cn,CT=ct,PG=pg,PL=pl,PW=pw,EOF[=yn7],P90=fv90,...,P99=fv99].

device   The element name of the terminal device being defined. This name can be one through seven characters long. The first character must be a letter, the other characters can be letters or digits. If this statement is used within the set following a GROUP statement, the device name cannot be longer than five characters. This is the name used by the host or NPU operator to monitor and control the device. This name is required; there is no default value.

All other parameters are described in the text.

Figure 6-5. TERMDEV Statement Format for Communication Lines of TIPTYPE=MODE4

TC PARAMETER

The TC parameter specifies the terminal class appropriate for terminal connections. Supported terminals are grouped into terminal classes, according to their hardware characteristics.

Each CDC-defined terminal class has the default characteristics associated with an archetype terminal. Each CDC-defined terminal class has a range of possible characteristics. These ranges determine the values allowed for parameters you specify on the TERMDEV or DEVICE statements.

Because the network cannot recognize differences among some terminal classes, a default terminal class exists for each subcategory of the TIP type (the subTIPtype, defined by the STIP parameter of the TERMINAL or TERMDEV statement). These defaults can be used when you declare a STIP parameter.

The mnemonics identifying the terminal classes are derived from the identification of the archetype terminals defining the classes, with the following exceptions: you can use TC28, TC29 TC30, and TC31 to identify site-defined terminal classes with no default characteristics, corresponding to reserved Network Access Method (NAM) terminal class numbers.

Because the characteristics of terminals in site-defined classes are unknown, TERMINAL, TERMDEV, or DEVICE statement parameter value ranges are checked against those valid for the STIP value specified. Any value declared for a terminal or device parameter is valid if the parameter and value conform to requirements for that STIP value.

The TC parameter is optional unless the STIP parameter is omitted. If TC is specified as CCP or as TC 28 through TC31, the STIP parameter must be specified. If TC is omitted, the STIP parameter must be specified. If STIP is omitted, the TC parameter must be specified and cannot have the value CCP or TC28 through TC31.

If the STIP value of M4A is declared or used, the following reserved words are valid values for the TC parameter:

CCP	Specifies that CCP is to provide the default terminal class appropriate for the subTIPtype; using this value is equivalent to omitting the parameter.
200UT	Identifies a terminal compatible with the CDC 200 User Terminal.
734	Identifies a terminal compatible with the CDC 734 series terminal.
TC28	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.
TC29	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.
TC30	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.

TC31	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

If the STIP value of M4C is declared or used, the following reserved words are valid values for the TC parameter:

711	Identifies a terminal compatible with the CDC 711 series terminal.
714	Identifies a terminal compatible with the CDC 714-10/20 series terminal.
714X	Identifies a terminal compatible with the CDC 714-30 series terminal.
TC28	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.
TC29	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.
TC30	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.
TC31	Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

If the TC parameter is specified with a value of TC28, TC29, TC30, or TC31, then CCP must also be modified to accept these values. If the appropriate support code for TC28, TC29, TC30, or TC31 is not added to CCP, these values will be considered invalid by CCP.

If you omit the TC parameter or specify the value CCP, the default is selected according to the STIP value declared. The defaults are:

STIP Value	TC Value Used
M4A	200UT
M4C	711

CA PARAMETER

The CA parameter specifies the hardware cluster address for all the devices accessing the line through one terminal. When the line is configured for automatic recognition of terminals, this parameter is optional; otherwise it is required.

When you define terminals on a fixed-configuration line, you must specify a unique value of the CA parameter for each terminal that can use the line. There is no default value for a cluster address.

If you supply this parameter for an automatic recognition line, the value need not be unique for each terminal defined on the line but must be unique for each terminal that simultaneously accesses the line. The network software performs an additional match while identifying the terminal in the configuration file; any terminal accessing the line must use a cluster address you specify or it cannot use the network.

. The CA parameter can have the following values:

AUTOREC

> Indicates that CCP should determine the cluster address during automatic recognition of the terminal (valid only if the AUTO parameter is used in the corresponding LINE or GROUP statement); using this value is equivalent to omitting the parameter.

70 thru 7F

> Indicates the hexadecimal cluster address of the terminal you are defining.

If you specify the value of AUTOREC, terminals with any cluster address valid for mode 4 terminals can access the network.

CSET PARAMETER

The CSET parameter specifies the code set of the terminal and character set of its console. This parameter is optional.

When you declare a value for CSET, it must be one of the following reserved words. For terminals in classes 200UT, 734, and TC28 through TC31, it can be:

> ASCII   ASCII code set. Some mode 4A terminals have selectable code sets but a fixed console character set; the character set can be ASCII or the CDC External BCD set.

> AUTOREC   Indicates that CCP should use the code and character set appropriate for the terminal class or determined by automatic recognition; using this value is equivalent to omitting the parameter.

> BCD   CDC External BCD code set. Some mode 4A terminals have selectable code sets but a fixed console character set; the character set can be ASCII or the CDC External BCD set.

> CSET15   Site-defined code and character set, identified within the network software as character set number 15.

For terminals in classes 711, 714, and 714X, the CSET value can be:

> ASCII   ASCII code and character set.

> AUTOREC   Indicates that CCP should use the code and character set appropriate for the terminal class or determined by automatic recognition; using this value is equivalent to omitting the parameter.

> CSET15   Site-defined code and character set, identified within the network software as character set number 15.

If you specify CSET for a terminal on an automatic recognition line, the network software performs an additional match while identifying the terminal in the configuration file. If you omit CSET, no additional match is performed for an automatic recognition line and the recognized value is used.

For fixed-configuration lines, the default is selected according to the STIP value declared or used. The defaults are:

STIP Value	CSET Value Used
M4A	ASCII
M4C	ASCII

Once established, the default cannot be changed.

EOF PARAMETER

The EOF parameter indicates whether the MODE 4 TIP sends an end-of-file indicator to line printer devices to mark the position in the data where an end-of-file was read from disk. The end-of-file indicator is an ESC V sequence.

Possible values for this parameter are:

> NO   Indicates that an end-of-file indicator is not sent by the MODE 4 TIP to line printer devices.

> YES   Indicates that an end-of-file indicator is sent by the MODE 4 TIP to line printer devices.

If you omit the EOF parameter, the default value of NO is used.

RIC PARAMETER

The RIC parameter indicates whether the terminal has restricted interactive capabilities. This parameter is optional.

A terminal can have restricted interactive capabilities for many reasons; for example, perhaps no console device actually exists and interactive dialog must occur through a pair of batch devices.

The effect this parameter has depends on the application program the terminal uses. For example, RBF does not output the READY prompt and does not require a GO command to enable any defined passive devices when RIC is specified.

Possible values for this parameter are:

> NO   Indicates that the terminal has full interactive capabilities.

> YES   Indicates that the terminal has limited interactive capabilities.

If you omit the RIC parameter, the default value of NO is used.

MODE 4 DEVICE DEFINITIONS

On a mode 4 communication line, a device definition is different from a terminal definition. A mode 4 terminal consists of one or more devices; terminal and device are not synonymous. Within the network, a mode 4 device is synonymous with a mode 4A station or with a mode 4C device.

You must provide one TERMDEV or DEVICE statement for each device that accesses the communication line through the terminal you are defining. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal. Figures 6-5 and 6-6 present the formats of these statements for devices on mode 4 protocol synchronous lines.

You can declare up to three DEVICE statements if the line uses a STIP value of M4A, or up to 15 DEVICE statements if the line uses a STIP value of M4C. Mode 4A terminals can have up to three DEVICE statements for site-defined device types, or one DEVICE statement for each of the following CDC-defined device types:

Card reader

Console

Line printer

Mode 4C terminals can have up to 15 DEVICE statements for site-defined device types, or up to 15 DEVICE statements for CDC-defined console devices, or up to seven DEVICE statements for CDC-defined line printer devices.

The following TERMDEV and DEVICE parameters provide device definition for the terminal:

ABL	DO	PW
AUTOCON	DT	P90 through P99
B1	ELO	SDT
B2	HD	TA
CN	HN	UBL
CT	LK	UBZ
DBL	PG	XBZ
DBZ	PL	
DI	PRI	

The DT parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

For console devices:

device: DEVICE[,DT=CON,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,]

    [UBL=upblim,XBZ=xmitsiz,AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],LK=yn5,]

    [B1=b1,B2=b2,CN=cn,CT=ct,ELO=elo,PG=pg,PL=pl,PW=pw,]

    [P90=fv90,...,P99=fv99].

For card readers:

device: DEVICE,DT=CR[,TA=trmaddr,UBZ=upbsize,DBL=dwnblim,UBL=upblim,PRI[=yn1],]

    [DI[=yn2],P90=fv90,...,P99=fv99].

For line printers:

device: DEVICE,DT=LP[,SDT=subdt,TA=trmaddr,DBZ=dwnlsiz,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [DO=devord,PRI[=yn1],DI[=yn2],PW=pw,PL=pl,P90=fv90,...,P99=fv99].

For site-defined devices:

device: DEVICE,DT=DT12[,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

    [DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,DO=devord,AUTOCON[=yn1],]

    [PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],LK=yn5,B1=b1,B2=b2,CN=cn,CT=ct,PG=pg,]

    [PL=pl,PW=pw,P90=fv90,...,P99=fv99].

device   The element name of the terminal device being defined. This name can be 1 through 7 characters long; the first character must be a letter, the other characters can be letters or digits. If this statement is used within the set following a GROUP statement, the device name cannot be longer than 5 characters. This is the name used by the host or NPU operator to monitor and control the device. This name is required; there is no default value.

All other parameters are described in the text.

Figure 6-6. DEVICE Statement Format for Communication Lines of TIPTYPE=MODE4

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

If you specify a CDC-defined device type, the other device definition parameters on the DEVICE or TERMDEV statement:

    Are required by the NDL processor where indicated in the text

    Have the predefined default shown in the text

    Have the predefined ranges shown in the text

    Are checked for uniqueness and compatibility

If you specify a site-defined device type, the other device definition parameters on the DEVICE statement:

    Are not required by the NDL processor

    Have no predefined default

    Are not checked for uniqueness

    Are not checked for compatibility with other parameters on the same statement

When you specify the DT parameter, the following values are valid:

CON	Identifies a CDC-defined console device.
CR	Identifies a CDC-defined card reader device.
LP	Identifies a CDC-defined line printer device.
DT12	Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

If you omit the DT parameter, the value of CON is used.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON

or DT of DT12). The ABL parameter has the following range of values:

$$1 \leq abl \leq 7$$

The value you declare should be greater than or equal to the downline block limit (DBL parameter value) of the terminal. Maximum line usage can be obtained by setting both DBZ and DBL so that 1 second of data is queued by CCP.

For console devices (DT of CON), the host queues abl - dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory but can reduce the number of times an application program is rolled out.

The default value for this parameter is 2 (DT of CON only).

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether CCP should automatically connect the console device and any associated batch devices to the selected host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

NO	Indicates that CCP should not attempt automatic connection.
YES	Indicates that CCP should attempt automatic connection.

If you omit the AUTOCON parameter, the default value of NO is used for a DT of CON. If you specify this parameter without a value, the value of YES is used.

B1 PARAMETER

The B1 parameter specifies the character to be used as a user break 1 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 1 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 1 message as a job step interrupt from the terminal user.

This parameter is optional; it is allowed only for devices of DT=CON or DT=DT12. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B2, CN, or CT parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B1 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	B1 Value Used	ASCII Character
M4A	3A	:
M4C	3A	:

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

B2 PARAMETER

The B2 parameter specifies the character to be used as a user break 2 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 2 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 2 message as a job step termination from the terminal user.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B1, CN, or CT parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B2 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	B2 Value Used	ASCII Character
M4A	29	)
M4C	29	)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

CN PARAMETER

The CN parameter specifies the character to be used to abort (cancel) an input message. When the terminal user enters this character as the last character on a line, the network software discards the last message transmitted from the terminal.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B1, B2, or CT parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CN parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	CN Value Used	ASCII Character
M4A	28	(
M4C	28	(

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a terminal definition command. When the terminal user enters this character as the first character on a line, CCP interprets the line as a command. Among other functions, CCP supports terminal definition commands to determine or change the values you have established for the following parameters:

B1	CT	PL
B2	HN	PW
CN	PG	TC

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B1, B2, or CN parameters

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the STIP parameter. The defaults are:

STIP Value	CT Value Used	ASCII Character
M4A	25	%
M4C	25	%

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between the host computer and this terminal. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following value range:

$$1 \leq dbl \leq 7$$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the terminal. Maximum line usage can be obtained by setting both DBZ and DBL so that 1 second of data is queued by CCP.

For console devices (DT of CON), the host queues abl - dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

For batch devices, the NPU queues all downline blocks. No ABL value exists, so the DBL value alone determines the impact of block queueing on NPU and host resources. Large DBL values rapidly use up NPU memory; small DBL values may not keep the terminal busy.

If you omit the DBL parameter, a default value is used. Default values depend on the device type and the terminal class. The default values are:

Terminal Class	DT Value	DBL Value Used
200UT and 734	CON	2
200UT and 734	CR	2
711, 714, 714X	CON	2
200UT and 734	LP	1
714 and 714X	LP	1

For terminals in classes 200UT and 734, a DBL value of 2 is recommended for printers (DT of LP) on lines that support speeds above 9600 bits per second.

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. Larger DBZ values cause fewer disk accesses by the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional; it is allowed for all device types except card readers (DT of CR). The DBZ parameter has the following range of values:

$$1 \leq dbz \leq 2043$$

The value you declare should be chosen together with the value used for the DBL parameter. For batch devices (DT of LP), you should use a value that is a multiple of 640 characters (one physical record unit). If dbz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used
$1 \leq dbz \leq 640$	640
$641 \leq dbz \leq 1280$	1280
$1281 \leq dbz \leq 2043$	1920

The NDL processor issues an informative message if it adjusts the DBZ value that you specify.

If you omit the DBZ parameter, a default value is used. Default values depend on the the STIP value declared or used and the device type. The default values are:

STIP Value	DT Value	DBZ Value Used
M4A	CON	1040
M4A	LP	640
M4C	CON	1280
M4C	LP	640

These defaults optimize use of the default value for the DBL parameter.

DI PARAMETER

The DI parameter is a stand-alone keyword that specifies a device as initially enabled or disabled. An enabled device is configured and serviced as soon as the communication line becomes active. A disabled device is neither configured nor serviced when the line becomes active.

If you initially disable the device, a host or NPU operator can change the device status to enabled. This change can be made only when the line becomes active (when a call is received on a dialup line, or when communications are established on a hard-wired line).

This parameter is optional. If you specify the DI parameter, you must use one of the following values:

NO   Indicates that the device is initially enabled.

YES Indicates that the device is initially disabled.

If you specify the DI parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

DO PARAMETER

The DO parameter specifies the device ordinal of the device within the terminal. This device ordinal provides a unique identifier to an application program for the device when more than one device of the same type is part of the terminal.

This parameter is allowed for line printers (DT of LP) or site-defined devices (DT of DT12) with a STIP value of M4C only. The DO parameter is optional if only one device of the same type is part of the terminal; otherwise, a unique value is required for each device you define.

If you specify this parameter, you must use a value in the range:

$$1 \leq devord \leq 7$$

If you omit this parameter when it is optional, the default value of 1 is used.

ELO PARAMETER

The ELO parameter indicates the event that identifies an end-of-line condition (an end-of-logical line or message-forwarding signal). During input, a message is transmitted upline as soon as an end-of-line code (or message-forwarding code sequence) or an end-of-block condition is detected.

When some devices perform input, they use code sequences or events to indicate the end-of-line and end-of-block conditions. The ELO parameter allows you to set the end-of-line indicator to one of these sequences or events.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP Indicates that CCP should use the default condition appropriate for the terminal class; using this value is equivalent to omitting the parameter.

EB   Indicates that the current end-of-block character code or code sequence for the terminal class should be the end-of-line condition indicator. This value causes any default end-of-line codes or code sequences to be sent upline as data within a single message.

If you omit this parameter or specify the value CCP, the end-of-line condition indicator is the normal end-of-line sequence for the device (DT of CON only).

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

HD PARAMETER

The HD parameter controls whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional and is valid only for console devices.

If you specify this parameter, you must use one of the following values:

NO   Indicates that the full host availability display is not presented to the terminal user. The terminal user receives only the host status message and the prompt message.

YES   Indicates that the host availability display should be presented to the terminal user.

If you omit this parameter and only a single host is defined, the default value of NO is used. If you omit this parameter and multiple hosts are defined, the default value of YES is used.

The value of this parameter does not change when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can also change the value you specify.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device and any associated batch devices are connected to unless another path is selected by the terminal user.

This parameter is optional and is valid only for console devices. If you declare this parameter, you must use one of the following values:

NONE

Indicates that no node number is specified. This has the same effect as not declaring the parameter. A value of NONE cannot be declared if AUTOCON is also specified and there are logical links to more than one host terminating in the NPU being defined.

$1 \leq hn \leq 255$

Indicates the node number of the host that the console device and any associated batch devices are to be connected to.

The value declared for the HN parameter must be the same as the NODE value in a COUPLER statement within the same network definition. If all the COUPLER statements for all the logical links to the NPU being defined have equal HNAME parameters, the NDL processor provides a default host node. The default host node is the last coupler specified. If the HN parameter is not specified, the terminal user must select a host before a connection can be made. If the AUTOCON parameter is specified, then the HN parameter must also be specified.

The value of this parameter does not change when the terminal class is changed from the console. The terminal user can change the value you specify.

LK PARAMETER

The LK parameter specifies whether unsolicited messages from the NPU or host operator can appear at the terminal. This parameter is optional; it is allowed only for console devices and site-defined devices (DT of CON or DT of DT12).

If you declare this parameter, you must specify one of the following values:

CCP   Indicates that CCP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that unsolicited messages should be delivered to the terminal as soon as received by CCP.

YES   Indicates that unsolicited messages should be discarded (locked out).

If you omit this parameter or specify the value CCP, the default value of NO is used (DT of CON only). The terminal user or the application program can change any value that you specify.

PG PARAMETER

The PG parameter specifies whether CCP should wait at each output page boundary for terminal user acknowledgment before it displays the next page of data. In certain situations, page waiting can occur other than at page boundaries; CCP produces a prompting message (OVER..) when this type of page waiting occurs. The user's response to page waiting is entry of a line, usually empty.

CCP views a new page as beginning at the start of each downline message. CCP calculates the length of a page from the current values of the page width and page length (see PW and PL Parameter descriptions). If the page width is infinite (PW=0), a page consists of one line less than the number of logical lines specified as the page length. If the page width is finite (PW is nonzero), a page consists of one line less than the number of physical lines specified by the page length; CCP calculates the number of physical lines by dividing each logical line into units less than or equal to the page width.

This parameter is optional; it is allowed only for console devices and site-defined devices (DT of CON or DT of DT12). If you specify this parameter, you must use one of the following reserved values:

CCP   Indicates that CCP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that page waiting should not occur.

YES   Indicates that page waiting should occur.

When you specify PG=YES, you should also specify a nonzero value for the PL parameter. CCP cannot perform page waiting at the boundaries of infinitely long pages (PL=0).

If you omit the PG parameter or specify a value of CCP, the default value of YES is used.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

PL PARAMETER

The PL parameter specifies the number of physical lines per page of output for the device. If the device is configured for page waiting (PG parameter), any message containing more lines of output than the page length will be interrupted by CCP for a page waiting response from the terminal user. The interruption occurs after line pl - 1 is output.

This parameter is optional; it is allowed for console devices, remote batch line printers, and site-defined devices only (DT of CON, DT of LP, or DT of DT12). If you specify the PL parameter, you must use one of the following values:

CCP

Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

Indicates an infinite page length. An infinite page length means that no page waiting occurs.

$8 \leq pl \leq 255$

Indicates the number of physical lines per page.

If you omit this parameter or specify a value of CCP, a default value is used (DT of CON only). The

default value depends on the terminal class. The defaults are:

TC Value	PL Value Used
200UT	13
734	13
711	16
714	16
714X	16

The NDL default value of the PL parameter for DT of LP is 64.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

PRI PARAMETER

The PRI parameter is a stand-alone keyword that indicates whether data to or from the device is to have traffic priority over that to or from other devices in the network. Data to or from a device that does not have traffic priority is suspended to relieve network congestion before data to or from terminals with traffic priority is suspended. Interactive devices should usually be given traffic priority over batch devices.

This parameter is optional. If you specify the PRI parameter, you must use one of the following values:

NO   Indicates that the device should not have data traffic priority.

YES   Indicates that the device should have data traffic priority.

If you declare this parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

PW PARAMETER

The PW parameter defines the number of characters per physical line of input or output for this device. This physical line length is also called the page width.

For console devices, output lines longer than pw characters are not divided but are counted as more than one line for page width calculations. If the PW value for the device is inappropriate, loss of visual fidelity can occur; the application program might divide the data into lines that are too short or too long for the screen's capacity, instead of allowing the terminal to wrap lines when needed.

This parameter is optional. If you specify the PW parameter, the value you can use depends on the device type and the terminal class. The following values are allowed for console devices and site-

defined devices (DT of CON or DT of DT12) in ter-
minal classes 200UT, 734, 711, 714, and 714X, and
for all devices in terminal classes TC28 through
TC31:

CCP

> Indicates that CCP should use the default
> value appropriate for the terminal class;
> using this value is equivalent to omitting
> the parameter.

0

> Indicates that the device has an infinite
> page width. An infinite page width means
> that physical line length has no effect on
> output formatting.

$50 \leq pw \leq 255$

> Indicates that the device can support
> physical lines no longer than the indicated
> value.

The following values are allowed for line printers
(DT of LP) in terminal classes 200UT, 734, 714, and
714X:

CCP

> Indicates that CCP should use the default
> value appropriate for the terminal class;
> using this value is equivalent to omitting
> the parameter.

$50 \leq pw \leq 255$

> Indicates that the device can support
> physical lines no longer than the indicated
> value.

For line printers, pw provides RBF with an indica-
tion of the character bytes that can comprise a
physical record of output (the number of characters
in a printer line).

If you omit this parameter or specify the value
CCP, a default value is used. The default value
depends on the device type. Default values are:

DT Value	PW Value Used
CON	80
LP	136

The value of this parameter for the console reverts
to the default value when the terminal class is
changed from the console or changed by the applica-
tion program. The application program can change
any value you specify for any device; the terminal
user can change any value you specify for the
console.

P90 THROUGH P99 PARAMETERS

These ten parameters indicate the hexadecimal field
value to use for the corresponding field number
within the Terminal Interface Program configuration
information transmitted for this device. The

released version of the CDC-written MODE4 TIP does
not use any of these ten field number/field value
pairs.

These parameters are optional. If you specify a
value, it must be within the following range:

$0 \leq fv9i \leq FF$

If you omit these parameters, there are no default
values.

SDT PARAMETER

The SDT parameter specifies the subdevice type of
the device you are defining. The subdevice type is
the set of external characteristics of interest to
the network software. This parameter is optional
and valid only for devices with DT values of LP and
DT12. The recognized reserved word values you can
use and the defaults depend on the value you
specify or use for the DT parameter.

If a DT value of LP is used, the following values
are allowed definitions of a printer character set:

A6
> Indicates that the device uses 64 ASCII
> characters.

A9
> Indicates that the device uses 95 ASCII
> characters.

B6
> Indicates that the device uses the
> 64-character CDC scientific (BCD)
> character set.

CCP
> Indicates that CCP should use the
> default appropriate for the device;
> using this value is equivalent to
> omitting the parameter.

SDT12
> Indicates a site-defined subdevice type
> that uses the subdevice type number 12.

SDT13
> Indicates a site-defined subdevice type
> that uses the subdevice type number 13.

SDT14
> Indicates a site-defined subdevice type
> that uses the subdevice type number 14.

SDT15
> Indicates a site-defined subdevice type
> that uses the subdevice type number 15.

If you omit the SDT parameter or specify the value
CCP, the default value of A6 is used.

If a DT value of CR is used, the following values
are allowed definitions of subdevice types:

SDT12
> Indicates a site-defined subdevice type
> that uses the subdevice type number 12.

SDT13
> Indicates a site-defined subdevice type
> that uses the subdevice type number 13.

SDT14
> Indicates a site-defined subdevice type
> that uses the subdevice type number 14.

SDT15
> Indicates a site-defined subdevice type
> that uses the subdevice type number 15.

If a DT value of DT12 is used, the following values are allowed definitions of subdevice type:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

26   Indicates that the device uses the 026 punch pattern set.

29   Indicates that the device uses the 029 punch pattern set.

6BIT   Indicates that the device uses a 6-bit byte for binary instruction codes.

8BIT   Indicates that the device uses an 8-bit byte for binary instruction codes.

If you omit the SDT parameter, no default exists.

TA PARAMETER

The TA parameter defines the terminal address of the device within the terminal cluster. This parameter must match the code actually used to communicate with the device.

The TA parameter is optional for terminals on automatic recognition lines but required for terminals on fixed-configuration lines. Any value used must be unique for mode 4C devices on the same terminal.

If you specify this parameter, you must use one of the following values:

AUTOREC

    Indicates that CCP should determine the terminal address code during automatic recognition of the terminal (valid only if the AUTO parameter is used in the corresponding LINE or GROUP statement); using this value is equivalent to omitting the parameter.

60

    Indicates the hexadecimal terminal address of a mode 4A terminal device.

61 thru 6F

    Indicates the hexadecimal terminal address of a mode 4C terminal device.

If you omit this parameter or specify the value AUTOREC, the value determined by automatic recognition of the line is used.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input message blocks and if you choose a UBL value smaller than this number, the application program becomes dead-locked (it waits for blocks of data that CCP is forced to discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following range of values:

$1 \leq ubl \leq 31$

If you omit the UBL parameter, a default value is used. The default value is 7.

The value you specify can be changed by the application program for consoles.

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline block can contain. CCP divides each message from the terminal into blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host.

You should choose a value that allows most of the logical lines entered by the device to fit into a single block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional; it is allowed for all device types except line printers (DT of LP). The UBZ parameter has the following range of values for console devices (DT of CON):

$0 \leq ubz \leq 2000$

The range for batch devices (DT of CR) is:

$1 \leq ubz \leq 2043$

The range for site-defined devices (DT of DT12) is:

$0 \leq ubz \leq 2043$

If you specify 0 for a console device, CCP sends an upline block whenever it receives 100 characters.

The value you declare should be chosen together with the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes. For console devices (DT of CON), rounding occurs as follows:

Value Supplied	Value Used in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

For batch devices (DT of CR), you should use a value that is a multiple of 640 characters (one physical record unit). If ubz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used
$1 \leq ubz \leq 640$	640
$641 \leq ubz \leq 1280$	1280
$1281 \leq ubz \leq 2043$	1920

The NDL processor issues an informative message if it adjusts the UBZ value that you specify.

If you omit the UBZ parameter, a default value is used. Default values depend on the device type. The defaults are:

DT Value	UBZ Value Used
CON	100
CR	640

These defaults optimize use of the default value for the UBL parameter. These default values are the recommended values.

For the Message Control System (MCS), the upline block size must be set to 0.

The application program can change the value you specify. The terminal user can change the value for consoles within the following range:

$$0 \leq ubz \leq 200$$

XBZ PARAMETER

The transmission block size (XBZ) parameter specifies the maximum number of character bytes each block sent to the terminal can contain. CCP divides downline blocks as necessary to create a block of the specified number of characters.

The transmission block size for a mode 4 console device should be the same as the number of characters that fit on its screen. A mode 4A console normally has a screen that is 80 characters by 13 lines; a mode 4C console normally has a screen that is 80 characters by 16 lines. The value you use should be less than or equal to the size of any buffer memory within the terminal.

This parameter is optional. The XBZ parameter has the following range of values:

$$200 \leq xbz \leq 2043$$

If you omit the XBZ parameter, a default value is used. Default values depend on the STIP value declared or used, the terminal class declared or used, and the device type. The defaults for console devices (DT of CON) in terminal classes 200UT, 734, 711, 714, and 714X are:

STIP Value	XBZ Value Used
M4A	1040
M4C	1280

The defaults for batch devices (DT of LP) are:

TC Value	XBZ Value Used
200UT	1000
734	1000
714	240
714X	1280

These default values are based on the buffer sizes of the archetype terminals and should not be increased unless the devices used have larger buffers.

The application program can change the value you specify. The terminal user can change the value for consoles.

HASP TERMINAL DEFINITIONS

On a synchronous communication line, a terminal definition is different from a device definition. A synchronous terminal consists of one or more devices; terminal and device are not synonymous. In a CDC network, a terminal is equivalent to a HASP workstation.

If you defined the communication line as a fixed definition line, you can declare only one TERMINAL or TERMDEV statement for the line. If you defined the line as an automatic recognition line, you can declare up to 255 TERMINAL or TERMDEV statements for the line.

You must provide at least one TERMINAL or TERMDEV statement. Figures 6-7 and 6-8 present the formats of these statements for terminals on CDC-defined HASP protocol synchronous lines. The following parameters provide the terminal definition for the line:

CO	STIP
CSET	TC
RIC	

```
TERMINAL[,STIP=stiptyp,TC=trmclas,]

    [CSET=charset,RIC=yn1,]

    [CO=conord].


Parameters are described in the text.
```

Figure 6-7. TERMINAL Statement Format
for Communication Lines of TIPTYPE=HASP

The STIP and TC parameters are described first
because they are related and affect the valid-
ity, ranges, and defaults of the other param-
eters. The other parameters are described in
alphabetical order.

STIP PARAMETER

The STIP parameter specifies the subTIPtype
required to support the terminal definition.
This parameter is optional if the TC parameter
specifies a CDC-defined terminal class. If STIP
is omitted, the TC parameter must be specified
and cannot be CCP or TC28 through TC31.

When you declare a value for STIP, it must be
one of the following reserved words:

AUTOREC   Specifies that CCP should use the
          subTIPtype appropriate for the ter-
          minal class (valid only if TC is
          specified); using this value is equiv-
          alent to omitting the parameter.

POST   Identifies a HASP terminal that only
          supports postprint format control.

PRE   Identifies a HASP terminal that sup-
          ports preprint format control.

If you use the AUTOREC value or omit the STIP
parameter, the default is selected according to the
terminal class declared. The defaults are:

TC Value	STIP Value Used
HASP	POST
HPRE	PRE

TC PARAMETER

The TC parameter specifies the terminal class
appropriate for the terminal. Supported devices
are grouped into terminal classes, according to
their hardware characteristics.

Each CDC-defined terminal class has the default
characteristics associated with an archetype ter-
minal. Each CDC-defined terminal class has a range
of possible characteristics. These ranges deter-
mine the values allowed for parameters you specify
on the TERMDEV or DEVICE statements.

The mnemonics identifying the terminal classes are
derived from the identification of the archetype
terminals defining the classes, with the following
exceptions: you can use TC28, TC29, TC30, and TC31
to identify site-defined terminal classes with no
default characteristics, corresponding to reserved
Network Access Method (NAM) terminal class numbers.

Because the network cannot recognize differences
among some terminal classes, a default terminal
class exists for each subcategory of the TIP type
(the subTIPtype, defined by the STIP parameter of
the TERMINAL or TERMDEV statement); these defaults
can be used when you declare a STIP parameter.

```
For console devices:

device:  TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,RIC=yn1,CO=conord,DT=CON,ABL=abl,]

    [DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,AUTOCON[=yn2],]

    [PRI[=yn3],DI[=yn4],HN=node,HD[=yn5],LK=yn6,B1=b1,B2=b2,CN=cn,CT=ct,PW=pw,]

    [P90=fv90,...,P99=fv99].


For site-defined devices:

device:  TERMDEV,[STIP=stiptyp,TC=trmclas,CSET=charset,RIC=yn1,CO=conord,]DT=DT12[,]

    [SDT=subdt,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [DO=devord,STREAM=streamno,AUTOCON[=yn2],PRI[=yn3],DI[=yn4],HN=node,HD[=yn5],LK=yn6,]

    [B1=b1,B2=b2,CN=cn,CT=ct,PW=pw,P90=fv90,...,P99=fv99].


device    The element name of the terminal device being defined. This name can be one through seven
          characters long. The first character must be a letter; the other characters can be
          letters or digits. If this statement is used within the set following a GROUP statement,
          the device name cannot be longer than five characters. This is the name used by the host
          or NPU operator to monitor and control the device. This name is required; there is no
          default value.

All other parameters are described in the text.
```

Figure 6-8. TERMDEV Statement Format for Communication Lines of TIPTYPE=HASP

Because the characteristics of terminals in site-defined classes are unknown, TERMINAL, TERMDEV, or DEVICE statement parameter value ranges are checked against the STIP value specified. Any value declared for a terminal or device parameter is valid if the parameter and value conform to requirements for that subTIPtype.

The following reserved words are valid values for the TC parameter:

CCP   Specifies that CCP is to provide the default terminal class appropriate for the subTIPtype; using this value is equivalent to omitting the parameter.

HASP   Identifies a HASP terminal that only supports postprinting format control.

HPRE   Identifies a HASP terminal that supports preprinting format control.

TC28   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.

TC29   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.

TC30   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.

TC31   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

If the TC parameter is omitted, the STIP parameter must be specified. If TC is specified as CCP or TC28 through TC31, STIP must be specified. If STIP is omitted, the TC parameter must be specified and cannot have the values CCP or TC28 through TC31.

If the TC parameter is specified with a value of TC28, TC29, TC30, or TC31, then CCP must also be modified to accept these values. If the appropriate support code for TC28, TC29, TC30, or TC31 is not added to CCP, these values will be considered invalid by CCP.

If you omit the TC parameter or specify the value CCP, the default is selected according to the STIP value declared. The defaults are:

STIP Value	TC Value Used
POST	HASP
PRE	HPRE

CO PARAMETER

The CO parameter specifies the configuration ordinal for the terminal. When the line is configured for automatic recognition of terminals, this parameter is optional; otherwise it is not allowed.

The terminal user can enter the configuration ordinal during automatic recognition. This allows the user to select the appropriate set of terminal and device definitions among several you supply for the line.

If you supply this parameter for an automatic recognition line, the value must be unique for each terminal that can use the line. The network software checks this value while identifying a terminal in the configuration file; any terminal accessing the line must supply the configuration ordinal you specify or it cannot use the network.

The CO parameter can have the following values:

AUTOREC

Indicates that CCP should use any configuration ordinal transmitted during automatic recognition of the terminal (valid only if the AUTO parameter is specified); using this value is equivalent to omitting the parameter.

$1 \leq$ conord ≤ 255

Indicates the decimal configuration ordinal of the terminal you are defining.

If you specify the value AUTOREC, CCP accepts any configuration ordinal valid for HASP terminals.

CSET PARAMETER

The CSET parameter specifies the code and character set of the terminal. This parameter is optional.

When you declare a value for CSET, it must be one of the following reserved words:

AUTOREC The default code set for the terminal class or that is determined during automatic recognition of the terminal device is used; using this value is equivalent to omitting the parameter.

CSET15   Site-defined code and character set, identified within the network software as character set number 15.

EBCDIC   Indicates the IBM Extended Binary Coded Decimal Interchange Code set and character set.

If you specify CSET for a terminal on an automatic recognition line, the network software performs an additional match while identifying the terminal in the configuration file. If you omit CSET, no additional match is performed for an automatic recognition line and the recognized value is used for fixed-configuration lines.

The default depends on the TC value declared or used. The defaults are:

TC Value	CSET Value Used
HASP	EBCDIC
HPRE	EBCDIC

Once established, the default value cannot be changed.

RIC PARAMETER

The RIC parameter indicates whether the terminal has restricted interactive capabilities. This parameter is optional.

A terminal can have restricted interactive capabilities for many reasons; for example, perhaps no console device actually exists and interactive dialog must occur through a pair of batch devices.

The effect this parameter has depends on the application program the terminal uses. For example, RBF does not output the READY prompt and does not require a GO command to enable any defined passive devices when RIC is specified.

Possible values for this parameter are:

NO   Indicates that the terminal has full interactive capabilities.

YES   Indicates that the terminal has limited interactive capablities.

If you omit the RIC parameter, the default value of YES is used.

HASP DEVICE DEFINITIONS

On a HASP communication line, a device definition is different from a terminal definition. A HASP terminal consists of one or more devices; terminal and device are not synonymous. In a CDC network, a device is equivalent to an equipment input or output stream within a HASP workstation.

You must provide one TERMDEV or DEVICE statement for each device that accesses the communication line through the terminal you are defining. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal. Figures 6-8 and 6-9 present the formats of these statements for devices on HASP protocol lines.

For console devices:

device: DEVICE[,DT=CON,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

    [XBZ=xmitsiz,AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],LK=yn5,]

    [B1=b1,B2=b2,CN=cn,CT=ct,PW=pw,P90=fv90,...,P99=fv99].

For card punches and plotters (DT of CP or PL):

device: DEVICE,DT=devtyp[,SDT=subdt,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

    [XBZ=xmitsiz,DO=devord,STREAM=streamno,PRI[=yn1],DI[=yn2],PW=pw,]

    [P90=fv90,...,P99=fv99].

For card readers:

device: DEVICE,DT=CR[,SDT=subdt,UBZ=upbsize,DBL=dwnblim,UBL=upblim,DO=devord,]

    [STREAM=streamno,PRI[=yn1],DI[=yn2],PW=pw,P90=fv90,...,P99=fv99].

For line printers:

device: DEVICE,DT=LP[,SDT=subdt,DBZ=dwnlsiz,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [DO=devord,STREAM=streamno,PRI[=yn1],DI[=yn2],PW=pw,P90=fv90,...,P99=fv99].

For site-defined devices:

device: DEVICE,DT=DT12[,SDT=subdt,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,]

    [UBL=upblim,XBZ=xmitsiz,DO=devord,STREAM=streamno,AUTOCON[=yn1],PRI[=yn2],]

    [DI[=yn3],HN=node,HD[=yn4],LK=yn5,B1=b1,B2=b2,CN=cn,CT=ct,PW=pw,P90=fv90,...,P99=fv99].

device   The element name of the terminal device being defined. This name can be one through five characters long. The first character must be a letter; the other characters can be letters or digits. If this statement is used within the set following a GROUP statement, the device name cannot be longer than five characters. This is the name used by the host or NPU operator to monitor and control the device. This name is required; there is no default value.

All other parameters are described in the text.

Figure 6-9. DEVICE Statement Format for Communication Lines of TIPTYPE=HASP

For each terminal you define, you can declare up to 22 DEVICE statements. HASP terminals can have up to 22 DEVICE statements for site-defined device types, or one DEVICE statement for a CDC-defined console device and up to seven DEVICE statements each for CDC-defined line printer, card reader, and card punch or plotter devices.

The following DEVICE and TERMDEV parameters provide the device definition for the terminal:

ABL	DI	P90 through P99
AUTOCON	DO	SDT
B1	DT	STREAM
B2	HD	UBL
CN	HN	UBZ
CT	LK	XBZ
DBL	PRI	
DBZ	PW	

The DT parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

If you specify a CDC-defined device type, the other device definition parameters on the DEVICE or TERMDEV statement:

Are required by the NDL processor where indicated in the text

Have the predefined default shown in the text

Have the predefined ranges shown in the text

Are checked for uniqueness and compatibility

If you specify a site-defined device type, the other device definition parameters on the DEVICE statement:

Are not required by the NDL processor

Have no predefined default

Are not checked for uniqueness

Are not checked for compatibility with other parameters on the same statement

When you specify the DT parameter, the following values are valid:

CON   Identifies a CDC-defined console device.

CP   Identifies a CDC-defined card punch device.

CR   Identifies a CDC-defined card reader device.

DT12   Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

LP   Identifies a CDC-defined line printer device.

If you omit the DT parameter, the value of CON is used.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). The ABL parameter has the following range of values:

$$1 \leq abl \leq 7$$

The value you declare should be greater than or equal to the downline block limit (DBL parameter value) of the terminal. For console devices (DT of CON), the host queues abl - dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory but can reduce the number of times an application program is rolled out.

The default value for this parameter is 2 (DT of CON only).

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether CCP should automatically connect the console device and any associated batch devices to the selected host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

NO   Indicates that CCP should not attempt automatic connection.

YES   Indicates that CCP should attempt automatic connection.

If you omit the AUTOCON parameter, the default value of NO is used for a DT of CON. If you specify this parameter without a value, the value of YES is used.

B1 PARAMETER

The B1 parameter specifies the character to be used as a user break 1 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 1 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 1 message as a job step interrupt from the terminal user.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B2, CN, or CT parameters

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B1 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the TC parameter. The defaults are:

TC Value	B1 Value Used	ASCII Character
HASP	3A	:
HPRE	3A	:

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

B2 PARAMETER

The B2 parameter specifies the character to be used as a user break 2 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 2 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 2 message as a job step termination from the terminal user.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B1, CN, or CT parameters

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B2 parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the TC parameter. The defaults are:

TC Value	B2 Value Used	ASCII Character
HASP	29	)
HPRE	29	)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

CN PARAMETER

The CN parameter specifies the character to be used to abort (cancel) an input message. When the terminal user enters this character as the last character on a line, the network software discards the message when it is transmitted from the terminal.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B1, B2, or CT parameters

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CN parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the TC parameter. The defaults are:

TC Value	CN Value Used	ASCII Character
HASP	28	(
HPRE	28	(

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a terminal definition command. When the terminal user enters this character as the first character on a line, CCP interprets the line as a command. Among other functions, CCP supports terminal definition commands to determine or change one of the values you have established for the following parameters:

    B1   B2   CN   CT   HN   PW   TC

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

    Any value that you declare or use by default for the B1, B2, or CN parameters

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the TC parameter. The defaults are:

TC Value	CT Value Used	ASCII Character
HASP	25	%
HPRE	25	%

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between the host computer and this terminal. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following value range:

$$1 \leq dbl \leq 7$$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the terminal. Maximum line usage can be obtained by setting both DBZ and DBL so that 1 second of data is queued by CCP.

For console devices (DT of CON), the host queues abl − dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

For batch devices, the NPU queues all downline blocks. No ABL value exists, so the DBL value alone determines the impact of block queueing on NPU resources. Large DBL values rapidly use up NPU memory; small DBL values may not keep the device busy.

If you omit the DBL parameter, a default value is used. The default values depend on the device type. The default values are:

DT Value	DBL Value Used
CON	2
CR	2
CP	1
LP	1
PL	1

For card punches, line printers, and plotters (DT of CP, LP, and PL) using lines of 19200 bits per second, a DBL value of 2 is recommended; for these batch devices using lines of 56000 bits per second, a DBL value of 5 is recommended.

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. Larger DBZ values cause fewer disk accesses by the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional; it is allowed for all device types except card readers (DT of CR). The DBZ parameter has the following range of values:

$$1 \leq dbz \leq 2043$$

The value you declare should be chosen together with the value used for the DBL parameter. Maximum line usage can be obtained by setting both DBZ and DBL so that 1 second of data is queued by CCP.

For batch devices (DT of CP, PL, or LP), you should use a value that is a multiple of 640 characters (one physical record unit). If dbz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used
$1 \leq dbz \leq 640$	640
$641 \leq dbz \leq 1280$	1280
$1281 \leq dbz \leq 2043$	1920

The NDL processor issues an informative message if it adjusts the DBZ value that you specify.

For consoles (DT of CON) using lines at 19200 bits per second or higher speeds, a DBZ value of 800 is recommended.

If you omit the DBZ parameter, a default value is used. Default values depend on the device type. The defaults are:

DT Value	DBZ Value Used
CON	400
CP	640
LP	640
PL	640

These defaults optimize use of the default values for the DBL parameter.

DI PARAMETER

The DI parameter is a stand-alone keyword that specifies a device as initially enabled or disabled. An enabled device is configured and serviced as soon as the communication line becomes active. A disabled device is neither configured nor serviced when the line becomes active.

If you initially disable the device, a host or NPU operator can change the device status to enabled. This change can be made only when the line becomes active (when a call is received on a dialup line, or when communications are established on a hard-wired line).

This parameter is optional. If you specify the DI parameter, you must use one of the following values:

NO   Indicates that the device is initially enabled.

YES   Indicates that the device is initially disabled.

If you specify the DI parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

DO PARAMETER

The DO parameter specifies the device ordinal of the device within the terminal. This device ordinal provides a unique identifier to an application program for the device when more than one device of the same type is part of the terminal.

This parameter is allowed for batch devices (DT of CR, CP, LP, or PL) or site-defined devices (DT of DT12) only. The DO parameter is optional if only one device of the same type is part of the terminal; otherwise, a unique value is required for each device you define.

If you specify this parameter, you must use a value in the range:

$$1 \leq devord \leq 7$$

If you omit this parameter when it is optional, the default value of 1 is used.

HD PARAMETER

The HD parameter controls whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional and is valid only for console devices.

If you specify this parameter, you must use one of the following values:

NO   Indicates that the full host availability display is not presented to the terminal user. The terminal user receives only the host status message and the prompt message.

YES   Indicates that the host availability display should be presented to the terminal user.

If you omit this parameter and only a single host is defined, the default value of NO is used. If you omit this parameter and multiple hosts are defined, the default value of YES is used.

The value of this parameter does not change when the terminal class is changed from the console. The application program or the terminal user can also change the value you specify.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device and any associated batch devices are connected to unless another path is selected by the terminal user.

This parameter is optional and is valid only for console devices. If you declare this parameter, you must use one of the following values:

NONE

Indicates that no node number is speci-
fied. This has the same effect as not
declaring the parameter. A value of NONE
cannot be declared if AUTOCON is also
specified and there are logical links to
more than one host terminating in the NPU
being defined.

$1 \leq hn \leq 255$

Indicates the node number of the host that
the console device and any associated batch
devices are to be connected to.

The value declared for the HN parameter must be the
same as the NODE value in a COUPLER statement within
the same network definition. If all the COUPLER
statements for all the logical links to the NPU
being defined have equal HNAME parameters, the NDL
processor provides a default host node. The
default host node is the last coupler specified.
If the HN parameter is not specified, the terminal
user must select a host before a connection can be
made. If the AUTOCON parameter is specified, then
the HN parameter must also be specified.

The value of this parameter does not change when
the terminal class is changed from the console.
The terminal user can change the value you specify.

LK PARAMETER

The LK parameter specifies whether unsolicited
messages from the NPU or host operator can appear
at the terminal. This parameter is optional; it is
allowed only for console devices and site-defined
devices (DT of CON or DT of DT12).

If you declare this parameter, you must specify one
of the following values:

CCP Indicates that CCP should use the
   default choice appropriate for the
   terminal class; using this value is
   equivalent to omitting the parameter.

NO Indicates that unsolicited messages
   should be delivered to the terminal as
   soon as received by CCP.

YES Indicates that unsolicited messages
   should be discarded (locked out).

If you omit this parameter or specify the value
CCP, the default value of NO is used (DT of CON
only).

PRI PARAMETER

The PRI parameter is a stand-alone keyword that
indicates whether data to or from the device is to
have traffic priority over that to or from other
devices in the network. Data to or from a device
that does not have traffic priority is suspended to
relieve network congestion before data to or from
terminals with traffic priority is suspended.
Interactive devices should usually be given traffic
priority over batch devices.

This parameter is optional. If you specify the PRI
parameter, you must use one of the following values:

NO Indicates that the device should not
   have data traffic priority.

YES Indicates that the device should have
   data traffic priority.

If you declare this parameter without a value, the
value of YES is used. If you omit this parameter,
the default value of NO is used.

PW PARAMETER

The PW parameter defines the number of characters
per physical line of input or output for this
device. This physical line length is also called
the page width.

For console devices, output lines longer than pw
characters might be divided into lines of pw or
fewer characters each by the host application
program. If the PW value for the device is
inappropriate, loss of visual fidelity can occur.
This happens if the application program divides the
data into lines that are too short or too long for
the screen's capacity, instead of allowing the
terminal to wrap lines when needed.

For batch devices, pw tells CCP how many character
bytes comprise a physical record of output (the
number of bytes in a plotter record, or the number
of characters in a printer line, or columns in a
punch card).

This parameter is optional. The values you can
specify depend on the terminal class and the device
type. The following values are allowed for console
devices and site-defined devices (DT of CON and DT
of DT12) in all terminal classes and for all
devices in site-defined terminal classes:

CCP

  Indicates that CCP should use the default
  value appropriate for the terminal class;
  using this value is equivalent to omitting
  the parameter.

0

  Indicates that the device has an infinite
  page width. An infinite page width means
  that physical line length has no effect on
  output formatting.

$50 \leq pw \leq 255$

  Indicates that the device can support
  physical lines no longer than the indicated
  value.

The following values are allowed for line printers
(DT of LP) in terminal classes HASP and HPRE:

CCP

  Indicates that CCP should use the default
  value appropriate for the terminal class;
  using this value is equivalent to omitting
  the parameter.

$$50 \leq pw \leq 150$$

Indicates that the device can support physical lines no longer than the indicated value.

If you omit this parameter or specify the value CCP, a default value is used. The default value depends on the terminal class and device type. Default values for console devices and plotters (DT of CON or PL) are:

TC Value	PW Value Used
HASP	80
HPRE	80

Default values for line printers (DT of LP) in terminal classes HPRE and HASP are:

TC Value	PW Value Used
HASP	120
HPRE	120

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value you specify for console devices; the application program can change any value you specify for batch devices.

P90 THROUGH P99 PARAMETERS

These ten parameters indicate the hexadecimal field value to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this device. The released version of the CDC-written HASP TIP does not use any of these ten field number/field value pairs.

These parameters are optional. If you specify a value, it must be within the following range:

$$0 \leq fv9i \leq FF$$

If you omit these parameters, there are no default values.

SDT PARAMETER

The SDT parameter specifies the subdevice type of the device you are defining. The subdevice type is the set of external characteristics of interest to the network software. This parameter is optional and valid only for devices with DT values of CR, LP, PL, and DT12. The recognized reserved word values you can use and the defaults depend on the value you specify or use for the DT parameter.

If a DT value of LP is used, the following values are allowed definitions of a printer character set:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

CCP   Indicates that CCP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

If you omit the SDT parameter or specify the value CCP, the default value of A6 is used.

If a DT value of CR is used, the following values are allowed definitions of a card reader punch pattern set:

CCP   Indicates that CCP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

26   Indicates that the default punch pattern at the beginning of each job deck is the 026 pattern set.

29   Indicates that the default punch pattern at the beginning of each job deck is the 029 pattern set.

If you omit the SDT parameter or specify the value CCP, the default value of 29 is used.

If a DT value of CP is used, the following values are allowed definitions of subdevice type:

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

If a DT value of PL is used, the following values are allowed definitions of a plotter instruction byte size:

CCP   Indicates that CCP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

6BIT   Indicates that the device uses a 6-bit byte for each instruction.

8BIT   Indicates that the device uses an 8-bit byte for each instruction.

If you omit the SDT parameter or specify the value CCP, the default value of 6BIT is used.

If a DT value of DT12 is used, the following values are allowed definitions of subdevice type:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

26   Indicates that the device uses the 026 pattern set.

29   Indicates that the device uses the 029 pattern set.

6BIT   Indicates that the device uses a 6-bit byte for each instruction.

8BIT   Indicates that the device uses an 8-bit byte for each instruction.

If you omit the SDT parameter, no default exists.

STREAM PARAMETER

The STREAM parameter defines the stream number used within the terminal to address data to or from the device. This stream number must be unique for each device of the same type within the terminal. If the terminal has plotters, the stream number for each plotter must be unique among all plotters and card punches within the terminal.

The STREAM parameter is allowed for batch devices (DT of CP, CR, LP, and PL) or site-defined devices (DT of DT12) only. This parameter is required for batch devices on fixed-configuration lines and optional for devices on automatic recognition lines.

If you specify this parameter, you must use one of the following values:

AUTOREC

   Specifies that the stream number is to be determined by the network software when automatic recognition of the terminal occurs (valid only if the LINE or GROUP statement contains the AUTO parameter); using this value is equivalent to omitting the parameter.

$1 \leq$ streamno ≤ 7

   Indicates the stream number associated with the device within the terminal.

If you omit this parameter or specify the value AUTOREC, the stream number determined during automatic recognition of the line is used.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input messages and if you choose a UBL value smaller than this number, the application program becomes deadlocked (it waits for blocks of data that CCP is forced to discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following range of values:

$$1 \leq ubl \leq 31$$

If you omit the UBL parameter, a default value is used (except for site-defined devices). The default value is 7.

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline block can contain. CCP divides each message from the terminal into blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host.

You should choose a value that allows most of the logical lines entered by the device to fit into a single block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional; it is allowed for all device types except line printers, card punches, and plotters (DT of LP, CP, or PL). The UBZ parameter has the following value range for console devices (DT of CON):

$$0 \leq ubz \leq 2000$$

For site-defined devices (DT of DT12), the range is:

$$0 \leq ubz \leq 2043$$

For batch devices (DT of CR), the range is:

$$1 \leq ubz \leq 2043$$

If you specify 0 for a console device, CCP sends an upline block whenever it receives 100 characters.

The value you declare should be chosen after considering the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes. For console devices (DT of CON), rounding occurs as follows:

Value Supplied	Value Used in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

For batch devices (DT of CR), you should use a value that is a multiple of 640 characters (one physical record unit). If ubz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used
$1 \leq ubz \leq 640$	640
$641 \leq ubz \leq 1280$	1280
$1281 \leq ubz \leq 2043$	1920

The NDL processor issues an informative message if it adjusts the UBZ value that you specify.

If you omit the UBZ parameter, a default value is used. Default values depend on the device type. The defaults are:

DT Value	UBZ Value Used
CON	100
CR	640

These defaults optimize use of the default values for the UBL parameter. These default values are the recommended values with two exceptions. The first exception is for card readers using lines at 9600 b/s. In this case, a UBZ value of 1280 is recommended. The second exception is for card readers using lines at speeds above 9600 b/s. In this case, a UBZ value of 1920 is recommended.

For the Message Control System (MCS), the upline block size must be set to 0.

The value you specify can be changed by the application program for consoles. The terminal user can change the value within the following range:

$$0 \leq ubz \leq 200$$

XBZ PARAMETER

The transmission block size (XBZ) parameter specifies the maximum number of character bytes each block sent to the terminal can contain. CCP divides downline network data blocks as necessary to create a transmission block of the specified number of characters.

The transmission block size for a HASP console is normally 400. This value is used for terminals on lines of 9600 bits per second or less. For line speeds higher than 9600 bits per second, the terminal's buffer size is often increased to 800, so the XBZ value used should also be 800. You should always choose a value less than or equal to the size of any buffer memory within the terminal.

This parameter is optional. The XBZ parameter has the following value range:

$$200 \leq xbz \leq 2043$$

If you omit the XBZ parameter, a default value of 400 is used. This default value is based on the buffer sizes of the archetype terminals and should not be increased unless the devices used have larger buffers.

The value you specify can be changed by the application program for consoles.

BISYNCHRONOUS TERMINAL DEFINITIONS

On a bisynchronous communication line, a terminal definition is different from a device definition. A bisynchronous terminal consists of one or more devices and might not have a console; terminal and device are not synonymous.

If you defined the communication line as a fixed-configuration line, you can declare only one TERMINAL or TERMDEV statement for the line. If you defined the line as an automatic recognition line, you can declare up to 255 TERMINAL or TERMDEV statements for the line; only one terminal can access the line at a given time.

You must provide one TERMINAL or TERMDEV statement for each terminal that can access the communication line. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal. Figures 6-10 and 6-11 present the formats of these statements for terminals on bisynchronous protocol lines.

```
TERMINAL[,STIP=stiptyp,TC=trmclas,]

    [CSET=charset,RIC=yn1,]

    [CO=conord,BCF[=yn2],MREC=mrec].


Parameters are described in the text.
```

Figure 6-10. TERMINAL Statement Format for Communication Lines of TIPTYPE=BSC

The following TERMDEV and DEVICE parameters provide the terminal definition for the line:

BCF   RIC
CO   STIP
CSET   TC
MREC

The STIP and TC parameters are described first because they are related and affect the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

STIP PARAMETER

The STIP parameter specifies the subTIPtype required to support the terminal definition. This parameter is optional if the TC parameter specifies a CDC-defined terminal class. If STIP is omitted, the TC parameter must be specified and cannot be CCP or TC28 through TC31.

When you declare a value for STIP, it must be one of the following reserved words:

AUTOREC   Specifies that the subTIPtype is to be determined by the network software when automatic recognition of the terminal occurs (valid only if TC is specified); using this value is equivalent to omitting the parameter.

2780   Indicates that the terminal uses the 2780 protocol variant.

3780   Indicates that the terminal uses the 3780 protocol variant.

```
For console devices:

device:  TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,RIC=yn1,CO=conord,BCF[=yn2],]

    [MREC=rec,DT=CON,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [AUTOCON[=yn3],PRI[=yn4],DI[=yn5],HN=node,HD[=yn6],CT=ct,PW=pw,P90=fv90,...,P99=fv99].


For site-defined devices:

device:  TERMDEV,[STIP=stiptyp,TC=trmclas,CSET=charset,RIC=yn1,CO=conord,BCF[=yn2],]

    [MREC=rec,]DT=DT12[,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,]

    [UBL=upblim,XBZ=xmitsiz,AUTOCON[=yn3],PRI[=yn4],DI[=yn5],HN=node,HD[=yn6],CT=ct,PW=pw,]

    [P90=fv90,...,P99=fv99].


device      The element name of the terminal device being defined. This name can be one through seven
            characters long. The first character must be a letter, the other characters can be letters
            or digits. If this statement is used within the set following a GROUP statement, the
            device name cannot be longer than five characters. This is the name used by the host or
            NPU operator to monitor and control the device. This name is required; there is no default
            value.

All other parameters are described in the text.
```

Figure 6-11. TERMDEV Statement Format for Communication Lines of TIPTYPE=BSC

If you use the AUTOREC value or omit the STIP parameter, the default is selected according to the terminal class declared. The defaults are:

TC Value	STIP Value Used
2780	2780
3780	3780

TC PARAMETER

The TC parameter specifies the terminal class appropriate for the terminal. Supported devices are grouped into terminal classes, according to their hardware characteristics.

Each CDC-defined terminal class has the default characteristics associated with an archetype terminal. Each CDC-defined terminal class has a range of possible characteristics. These ranges determine the values allowed for parameters you specify on the TERMDEV or DEVICE statements.

The mnemonics identifying the terminal classes are derived from the identification of the archetype terminals defining the classes, with the following exceptions: you can use TC28, TC29, TC30, and TC31 to identify site-defined terminal classes with no default characteristics, corresponding to reserved Network Access Method (NAM) terminal class numbers.

Because the network cannot recognize differences among some terminal classes, a default terminal class exists for each subcategory of the TIP type (the subTIPtype, defined by the STIP parameter of the TERMINAL or TERMDEV statement); these defaults can be used when you declare a STIP parameter.

Because the characteristics of terminals in site-defined classes are unknown, TERMINAL, TERMDEV, or DEVICE statement parameter value ranges are checked against those valid for the STIP value specified. Any value declared for a terminal or device parameter is valid if the parameter and value conform to requirements for that subTIPtype.

The following reserved words are valid values for the TC parameter:

CCP   Specifies that CCP is to provide the default terminal class appropriate for the subTIPtype; using this value is equivalent to omitting the parameter.

TC28   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.

TC29   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.

TC30   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.

TC31   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

2780   Identifies a terminal that is compatible with the IBM 2780.

3780   Identifies a terminal that is compatible with the IBM 3780.

If the TC parameter is specified as CCP or TC28 through TC31, the STIP parameter must be specified. If the TC parameter is omitted, the STIP parameter must be specified. If STIP is omitted, the TC parameter must be specified and cannot have the value CCP or TC28 through TC31.

If the TC parameter is specified with a value of TC28, TC29, TC30, or TC31, then CCP must also be modified to accept these values. If the appropriate support code for TC28, TC29, TC30, or TC31 is not added to CCP, these values will be considered invalid by CCP.

If you omit the TC parameter or specify the value CCP, the default is selected according to the STIP value declared. The defaults are:

STIP Value	TC Value Used
2780	2780
3780	3780

BCF PARAMETER

The BCF parameter indicates whether the terminal supports compression of blanks in upline and downline data. Blank compression consists of deleting blanks that occur in multiples and replacing each multiple with a code set sequence that represents the number deleted.

When the network software compresses blanks in downline data, it does so for all devices within the terminal. When the network software receives a compressed blank sequence in upline data from any device within the terminal, it expands the data by replacing each sequence with the appropriate number of blanks.

This stand-alone parameter is optional and allowed only for terminals in classes 2780 and TC28 through TC30. If you specify this parameter, you must use one of the following values:

NO   Indicates that the terminal does not compress blanks in its input and cannot process compressed blanks received as output.

YES   Indicates that the terminal compresses blanks in its input and can process compressed blanks received as output.

If you specify this parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used (TC of 2780 only).

CO PARAMETER

The CO parameter specifies the configuration ordinal for the terminal. When the line is configured for automatic recognition of terminals, this parameter is optional; otherwise it is not allowed.

The terminal user can enter the configuration ordinal during automatic recognition. This allows the user to select the appropriate set of terminal and device definitions among several you supply for the line.

If you supply this parameter for an automatic recognition line, the value must be unique for each terminal that can use the line. The network software checks this value while identifying the terminal from the configuration file. Any terminal accessing the line must supply a configuration ordinal you specify or it cannot use the network.

The CO parameter can have the following values:

AUTOREC

Indicates that CCP should determine the configuration ordinal during automatic recognition of the terminal (valid only if the AUTO parameter is specified); using this value is equivalent to omitting the parameter.

$1 \leq$ conord ≤ 255

Indicates the decimal configuration ordinal of the terminal you are defining.

If you specify the value AUTOREC, CCP accepts any configuration ordinal valid for bisynchronous terminals.

CSET PARAMETER

The CSET parameter specifies the code and character set of the terminal. This parameter is optional.

When you declare a value for CSET, it must be one of the following reserved words. For terminals in classes TC28 through TC31, it can be:

AUTOREC Indicates that CCP should use the code set appropriate for the terminal class or determined by automatic recognition; using this value is equivalent to omitting the parameter.

CSET15 Site-defined code and character set, identified within the network software as character set number 15.

EBCDIC Indicates the IBM Extended Binary Coded Decimal Interchange Code set and character set.

If you specify CSET for a terminal on an automatic recognition line, the network software performs an additional match while identifying the terminal in

the configuration file. If you omit CSET, no additional match is performed for automatic recognition lines and the recognized value is used.

For fixed-configuration lines, the default is selected according to the TC value declared or used. The defaults are:

TC Value	CSET Value Used
2780	EBCDIC
3780	EBCDIC

Once established, the default cannot be changed.

MREC PARAMETER

The MREC parameter indicates the maximum number of records that each device within the terminal can receive in each transmission block. This number determines how the network software blocks data into a transmission block for output.

This parameter is optional and allowed for terminal classes 2780 or TC28 through TC31 only. If you specify this parameter, you must use one of the following values:

CCP

Indicates that CCP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$1 \leq$ mrec ≤ 7

Indicates the number of records required by the terminal in each block exchanged with the network.

If you omit this parameter or specify the value CCP, the default value of 2 is used. This default supports the standard version of the 2780 protocol.

RIC PARAMETER

The RIC parameter indicates whether the terminal has restricted interactive capabilities. A terminal can have restricted interactive capabilities for many reasons; for example, perhaps no console device actually exists and interactive dialog must occur through a pair of batch devices.

The effect this parameter has depends on the application program the terminal uses. For example, RBF does not output the READY prompt and does not require a GO command to enable any defined passive devices when RIC is specified.

This parameter is optional. Possible values are:

NO Indicates that the terminal has full interactive capabilities.

YES Indicates that the terminal has limited interactive capablities.

If you omit the RIC parameter, the default value of YES is used.

BISYNCHRONOUS DEVICE DEFINITIONS

On a bisynchronous communication line, a device definition is different from a terminal definition. A bisynchronous terminal consists of one or more devices; terminal and device are not synonymous.

You must provide one TERMDEV or DEVICE statement for each device that accesses the communication line through the terminal you are defining. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal. Figures 6-11 and 6-12 present the formats of these statements for devices on bisynchronous protocol lines.

For each terminal you define, you can declare up to four DEVICE statements. Bisynchronous terminals can have up to four DEVICE statements for site-defined device types, or one DEVICE statement each for a CDC-defined console, line printer, card reader, or card punch device.

If a terminal does not have a console, you must define a dummy console device for it. The network software requires a console to exist in all terminal definitions so that batch devices can be logged in.

The following parameters provide the definition for devices within the terminal:

ABL	DT	SDT
AUTOCON	HD	TA
CT	HN	UBL
DBL	PRI	UBZ
DBZ	PW	XBZ
DI	P90 through P9	

The DT parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

For console devices:

device: DEVICE[,DT=CON,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],CT=ct,PW=pw,P90=fv90,...,P99=fv99].

For card readers:

device: DEVICE,DT=CR[,SDT=subdt,UBZ=upbsize,DBL=dwnblim,UBL=upblim,PRI[=yn1],]

    [DI[=yn2],PW=pw,P90=fv90,...,P99=fv99].

For card punches:

device: DEVICE,DT=CP[,TA=trmaddr,DBZ=dwnlsiz,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [PRI[=yn1],DI[=yn2],PW=pw,P90=fv90,...,P99=fv99].

For line printers:

device: DEVICE,DT=LP[,SDT=subdt,TA=trmaddr,DBZ=dwnlsiz,DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,]

    [PRI[=yn1],DI[=yn2],PW=pw,P90=fv90,...,P99=fv99].

For site-defined devices:

device: DEVICE,DT=DT12[,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,]

    [UBL=upblim,XBZ=xmitsiz,AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],]

    [CT=ct,PW=pw,P90=fv90,...,P99=fv99].

device   The element name of the terminal device being defined. This name can be one through seven characters long. The first character must be a letter; the other characters can be letters or digits. If this statement is used within the set following a GROUP statement, the device name cannot be longer than five characters. This is the name used by the host or NPU operator to monitor and control the device. This name is required; there is no default value.

All other parameters are described in the text.

Figure 6-12. DEVICE Statement Format for Communication Lines of TIPTYPE=BSC

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

If you specify a CDC-defined device type, the other device definition parameters on the DEVICE or TERMDEV statement:

Are required by the NDL processor where indicated in the text

Have the predefined default shown in the text

Have the predefined ranges shown in the text

Are checked for uniqueness and compatibility

If you specify a site-defined device type, the other device definition parameters on the DEVICE statement:

Are not required by the NDL processor

Have no predefined default

Are not checked for uniqueness

Are not checked for compatibility with other parameters on the same statement

When you specify the DT parameter, the following values are valid:

CON   Identifies a CDC-defined console device.

CP   Identifies a CDC-defined card punch device.

CR   Identifies a CDC-defined card reader device.

DT12   Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

LP   Identifies a CDC-defined line printer device.

If you omit the DT parameter, the value of CON is used.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON

or DT of DT12). The ABL parameter has the following range of values:

$$1 \leq abl \leq 7$$

The value you declare should be greater than or equal to the downline block limit (DBL parameter value) of the terminal. For console devices (DT of CON), the host queues abl - dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory but can reduce the number of times an application program is rolled out.

The default value for this parameter is 2 (DT of CON only).

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether CCP should automatically connect the console device and any associated batch devices to the selected or defaulted host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

NO   Indicates that CCP should not attempt automatic connection.

YES   Indicates that CCP should attempt automatic connection.

If you omit the AUTOCON parameter, the default value of NO is used for a DT of CON. If you specify this parameter without a value, the value of YES is used.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a terminal definition command. When the terminal user enters this character as the first character on a line, CCP interprets the line as a command. Among other functions, CCP supports terminal definition commands to determine or change the values you have established for the following parameters:

CT   HN   PW   TC

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, the default value is used (DT of CON only). The default value depends on the value declared or used for the TC parameter. The defaults are:

TC Value	CT Value Used	ASCII Character
2780	25	%
3780	25	%

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value you specify.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be out-standing (unacknowledged) between the host computer and this terminal. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following range of values:

$1 \leq dbl \leq 7$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the terminal. For console devices (DT of CON), the host queues abl − dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

For batch devices, the NPU queues all downline blocks. No ABL value exists, so the DBL value alone determines the impact of block queueing on NPU and host resources. Large DBL values rapidly use up NPU memory; small DBL values may not keep the device busy.

If you omit the DBL parameter, a default value is used. Default values depend on the device type. The defaults are:

DT Value	DBL Value Used
CON	2
CR	2
CP	1
LP	1

For card punches and line printers using lines at speeds above 9600 bits per second, a DBL value of 2 is recommended.

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. Larger DBZ values cause fewer disk accesses by the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional; it is allowed for all device types except card readers (DT of CR). The DBZ parameter has the following range of values:

$1 \leq dbz \leq 2043$

The value you declare should be chosen together with the value used for the DBL parameter. For batch devices (DT of CP or LP), you should use a value that is a multiple of 640 characters (one physical record unit). If dbz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used
$1 \leq dbz \leq 640$	640
$641 \leq dbz \leq 1280$	1280
$1281 \leq dbz \leq 2043$	1920

If you omit the DBZ parameter, a default value is used. Default values depend on the STIP value declared or used and the device type. The defaults are:

STIP Value	DT Value	DBZ Value Used
2780	CON	400
2780	CP	640
2780	LP	640
3780	CON	512
3780	CP	640
3780	LP	640

These defaults optimize use of the default values for the DBL parameter.

DI PARAMETER

The DI parameter is a stand-alone keyword that specifies a device as initially enabled or disabled. An enabled device is configured and serviced as soon as the communication line becomes active. A disabled device is neither configured nor serviced when the line becomes active.

If you initially disable the device, a host or NPU operator can change the device status to enabled. This change can be made only when the line becomes active (when a call is received on a dialup line, or when communications are established on a hard-wired line).

This parameter is optional. If you specify the DI parameter, you must use one of the following values:

NO   Indicates that the device is initially enabled.

YES   Indicates that the device is initially disabled.

If you specify the DI parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

HD PARAMETER

The HD parameter controls whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional and is valid only for console devices.

If you specify this parameter, you must use one of the following values:

NO   Indicates that the full host availability display is not presented to the terminal user The terminal user receives only the host status message and the prompt message.

YES   Indicates that the host availability display should be presented to the terminal user.

If you omit this parameter and only a single host is defined, the default value of NO is used. If you omit this parameter and multiple hosts are defined, the default value of YES is used.

The value of this parameter does not change when when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can also change the value you specify.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device and any associated batch devices are connected to unless another path is selected by the terminal user.

This parameter is optional and is valid only for console devices. If you declare this parameter, you must use one of the following values:

NONE

     Indicates that no node number is specified. This has the same effect as not declaring the parameter. A value of NONE cannot be declared if AUTOCON is also specified and there are logical links to more than one host terminating in the NPU being defined.

$1 \leq hn \leq 255$

     Indicates the node number of the host that the console device and any associated batch devices are to be connected to.

The value declared for the HN parameter must be the same as the NODE value in a COUPLER statement within the same network definition. If all the COUPLER statements for all the logical links to the NPU being defined have equal HNAME parameters, the NDL processor provides a default host node. The default host node is the last coupler specified. If the HN parameter is not specified, the terminal user must select a host before a connection can be made. If the AUTOCON parameter is specified, then the HN parameter must also be specified.

The value of this parameter does not change when the terminal class is changed from the console. The terminal user can change the value you specify.

PRI PARAMETER

The PRI parameter is a stand-alone keyword that indicates whether data to or from the device is to have traffic priority over that to or from other devices in the network. Data to or from a device that does not have traffic priority is suspended to relieve network congestion before data to or from terminals with traffic priority is suspended. Interactive devices should usually be given traffic priority over batch devices.

This parameter is optional. If you specify the PRI parameter, you must use one of the following values:

NO   Indicates that the device should not have data traffic priority.

YES   Indicates that the device should have data traffic priority.

If you declare this parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

PW PARAMETER

The PW parameter defines the number of characters per physical line of output for this device. This physical line length is also called the page width.

For console devices (DT of CON) with a printer defined as the output mechanism (OP of PR), output lines longer than pw characters are divided into lines of pw or fewer characters each.

This parameter is optional. The values you can use depend on the terminal class and the device type. The following values are allowed for console devices and site-defined devices (DT of CON or DT of DT12) in terminal classes 2780 and 3780 and for all devices in terminal classes TC28 through TC31:

CCP

     Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

     Indicates that the device has an infinite page width. An infinite page width means that physical line length has no effect on output formatting.

$$50 \leq pw \leq 255$$

> Indicates that the device can support physical lines no longer than the indicated value.

The following values are allowed for line printers (DT of LP) in terminal classes 2780 and 3780:

CCP

> Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$$50 \leq pw \leq 150$$

> Indicates that the device can support physical lines no longer than the indicated value.

If you omit this parameter or specify the value CCP, a default value is used. The default value depends on the terminal class and device type. Default values for console devices (DT of CON) are:

TC Value	PW Value Used
2780	80
3780	80

Default values for line printers (DT of LP) in terminal classes 2780 and 3780 are:

TC Value	PW Value Used
2780	80
3780	120

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program can change any value you specify; the terminal user can change any value you specify for a console.

P90 THROUGH P99 PARAMETERS

These ten parameters indicate the hexadecimal field value to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this device. The released version of the CDC-written BSC TIP does not use any of these ten field number/field value pairs.

These parameters are optional. If you specify a value, it must be within the following range:

$$0 \leq fv9i \leq FF$$

If you omit these parameters, there are no default values.

SDT PARAMETER

The SDT parameter specifies the subdevice type of the device you are defining. The subdevice type is the set of external characteristics of interest to the network software. This parameter is optional

and valid for devices with DT values of CR, LP, and DT12. The recognized reserved word values you can use and the defaults depend on the value you specify or use for the DT parameter.

If a DT value of LP is used, the following values are allowed definitions of a printer character set:

A6
> Indicates that the device uses 64 ASCII characters.

A9
> Indicates that the device uses 95 ASCII characters.

B6
> Indicates that the device uses the 64-character CDC scientific (BCD) character set.

CCP
> Indicates that CCP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12
> Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13
> Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14
> Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15
> Indicates a site-defined subdevice type that uses the subdevice type number 15.

If you omit the SDT parameter or specify the value CCP, the default value of A6 is used.

If a DT value of CR is used, the following values are allowed definitions of a card reader punch pattern set:

CCP
> Indicates that CCP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12
> Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13
> Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14
> Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15
> Indicates a site-defined subdevice type that uses the subdevice type number 15.

26
> Indicates that the default punch pattern at the beginning of each job deck is the 026 pattern set.

29
> Indicates that the default punch pattern at the beginning of each job deck is the 029 pattern set.

If you omit the SDT parameter or specify the value CCP, the default value of 29 is used.

If a DT value of CP is used, the following values are allowed definitions of subdevice type:

SDT12
> Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

If a DT value of DT12 is used, the following values are allowed definitions of subdevice type:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

26   Indicates that the device uses the 026 pattern set.

29   Indicates that the device uses the 029 pattern set.

6BIT   Indicates that the device uses a 6-bit byte for binary instruction codes.

8BIT   Indicates that the device uses an 8-bit byte for binary instruction codes.

If you omit the SDT parameter, no default exists.

TA PARAMETER

The TA parameter defines the addressing code of the card punch device within the terminal. This parameter must match the code actually used by the terminal as a flag for the routing of output data to a card punch.

The TA parameter is optional for terminals on automatic recognition lines but required for terminals on fixed-configuration lines. This parameter is valid only for card punches and site-defined devices (DT of CP or DT of DT12) in terminal classes 3780 and TC28 through TC31.

If you specify this parameter, you must use one of the following values:

AUTOREC   Indicates that CCP should determine the addressing code during automatic recognition of the terminal (valid only if the AUTO parameter is used in the corresponding LINE or GROUP statement); using this value is equivalent to omitting the parameter.

2   Indicates that the addressing code is the equivalent of the ASCII DC2 character.

3   Indicates that the addressing code is the equivalent of the ASCII DC3 character.

If you omit this parameter or specify the value AUTOREC, the value determined by automatic recognition of the line is used.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input messages and if you choose a UBL value smaller than this number, the application program becomes deadlocked (it waits for blocks of data that CCP is forced to discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following range of values:

$1 \leq ubl \leq 31$

If you omit the UBL parameter, a default value is used (except for site-defined devices). The default value is 7.

The value you specify can be changed by the application program for consoles.

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline network data block can contain. CCP divides each message from the terminal into blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host.

You should choose a value that allows most of the logical lines entered by the device to fit into a single block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional; it is allowed for all device types except line printers and card punches (DT of LP or CP). The UBZ parameter has the following range of values for console devices (DT of CON):

$0 \leq ubz \leq 2000$

For site-defined devices (DT of DT12), the range is:

$0 \leq ubz \leq 2043$

For batch devices (DT of CR), the range is:

$1 \leq ubz \leq 2043$

If you specify 0 for a console device, CCP sends an upline block whenever it receives 100 characters.

The value you declare should be chosen after considering the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes. For console devices (DT of CON), rounding occurs as follows:

Value Supplied	Value Used in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

For batch devices (DT of CR), you should use a value that is a multiple of 640 characters (one physical record unit). If ubz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used in File
$1 \leq ubz \leq 640$	640
$641 \leq ubz \leq 1280$	1280
$1281 \leq ubz \leq 2043$	1920

The NDL processor issues an informative message if it alters the value you supply.

If you omit the UBZ parameter, a default value is used. Default values depend on the device type. The defaults are:

DT Value	UBZ Value Used
CON	100
CR	640

These defaults optimize use of the default values for the UBL parameter. The default values are the recommended values with two exceptions. The first exception is for card readers using lines at 9600 b/s. In this case, a UBZ value of 1280 is the recommended value. The second exception is for card readers using lines at speeds above 9600 b/s. In this case, a UBZ value of 1920 is recommended.

For the Message Control System (MCS), the upline block size must be set to 0.

The application program can change the value you specify. The terminal user can change the value for consoles within the following range:

$0 \leq ubz \leq 200$

XBZ PARAMETER

The transmission block size (XBZ) parameter specifies the maximum number of character bytes each block sent to the terminal can contain. CCP divides downline network data blocks as necessary to create a transmission block of the specified number of characters.

The transmission block size for a dummy console device should be less than or equal to the XBZ value declared or used for its line printer. A bisynchronous terminal normally does not have a console; the console device configured as part of the terminal is actually a special input/output mode for the card reader and line printer. The value you choose must be less than or equal to the size of any buffer memory within the terminal.

This parameter is optional. The XBZ parameter has the following range of values:

$200 \leq xbz \leq 2043$

If you omit the XBZ parameter, a default value is used. Default values depend on the STIP value declared or used. The defaults (DT of CON, CP, and LP only) are:

STIP Value	XBZ Value Used
2780	400
3780	512

These default values are based on the buffer sizes of the archetype terminals and should not be increased unless the devices used have larger buffers.

The value you specify can be changed by the application program for consoles.

3270 TERMINAL DEFINITIONS

On a 3270 communication line, a terminal definition is different from a device definition. A 3270 terminal consists of one or more devices; terminal and device are not synonymous.

The 3270 communication line can only be defined as a fixed-configuration line. You can declare up to 32 TERMINAL or TERMDEV statements for the line.

You must provide one TERMINAL or TERMDEV statement for each terminal that can access the communication line. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal. Figures 6-13 and 6-14 present the formats of these statements for terminals on 3270 protocol lines.

The following TERMINAL and TERMDEV parameters provide the terminal definition for the line.

CA
CSET
RIC
TC

The TC parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

TC PARAMETER

The TC parameter specifies the terminal class appropriate for the terminal. Supported devices are grouped into terminal classes, according to their hardware characteristics.

```
TERMINAL, TC=trmclas,CA=clsadrC,CSET=charset,RIC=yn1].
```

Figure 6-13. TERMINAL Statement Format for Communication Lines of TIPTYPE=3270

```
For console devices:

device:  TERMDEVC,TC=trmclas,CSET=charset,CA=clsadr,RIC=yn1,DT=CON,TA=trmaddr,]

    CABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

    CAUTOCONC=yn2],PRIC=yn3],DIC=yn4],HN=node,HDC=yn5],LK=yn6,B1=b1,B2=b2,CN=cn,]

    CCT=ct,PG=pg,PL=pl,PW=pw].


For site-defined devices:

device:  TERMDEVC,TC=trmclas,CSET=charset,CA=clsadr,RIC=yn1],DT=DT12C,SDT=subdt,]

    CTA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,UBL=upblim,]

    CAUTOCONC=yn2],PRIC=yn3],DIC=yn4],HN=node,HDC=yn5],LK=yn6,B1=b1,]

    CB2=b2,CN=cn,CT=ct,PG=pg,PL=pl,PW=pw].


device      The element name of the terminal device being defined.  This name can be one through seven
            characters long.  The first character must be a letter, the other characters can be
            letters or digits.  If this statement is used within the set following a GROUP statement,
            the device name cannot be longer than five characters.  This is the name used by the host
            or NPU operator to monitor and control the device.  This name is required; there is no
            default value.

All other parameters are described in the text.
```

Figure 6-14. TERMDEV Statement Format for Communication Lines of TIPTYPE=3270

Each CDC-defined terminal class has the default characteristics associated with an archetype terminal. Each CDC-defined terminal class has a range of possible characteristics. These ranges determine the values allowed for parameters you specify on the TERMDEV or DEVICE statements.

The mnemonics identifying the terminal classes are derived from the identification of the archetype terminals defining the classes.

The TC parameter is required. When you declare a value for TC, it must be the following reserved word:

3270   Identifies a terminal that is compatible with the IBM 3270.

CA PARAMETER

The CA parameter specifies the hardware cluster address for all the devices accessing the line through one terminal. This parameter is required. You must specify a unique value of the CA parameter for each terminal that can use the line. There is no default value for a cluster address.

The CA parameter can have the following range of values:

0 through 1F

Indicates the hexadecimal cluster address of the terminal you are defining.

Cluster addresses are mapped to the following polling addresses:

CA Value	EBCDIC Hex	CA Value	EBCDIC Hex
0	40	10	50
1	C1	11	D1
2	C2	12	D2
3	C3	13	D3
4	C4	14	D4
5	C5	15	D5
6	C6	16	D6
7	C7	17	D7
8	C8	18	D8
9	C9	19	D9
A	4A	1A	5A
B	4B	1B	5B
C	4C	1C	5C
D	4D	1D	5D
E	4E	1E	5E
F	4F	1F	5F

CSET PARAMETER

The CSET parameter specifies the code and character set of the terminal. This parameter is optional.

When you declare a value for CSET, it must be the following reserved word:

EBCDIC   Indicates the IBM Extended Binary Coded Decimal Interchange Code set and character set.

RIC PARAMETER

The RIC parameter indicates whether the terminal has restricted interactive capabilities. A terminal can have restricted interactive capabilities for many reasons; for example, perhaps no console device actually exists and interactive dialog must occur through a pair of batch devices.

The effect this parameter has depends on the application program the terminal uses. For example, RBF does not output the READY prompt and does not require a GO command to enable any defined passive devices when RIC is specified.

This parameter is optional. Possible values are:

NO   Indicates that the terminal has full interactive capabilities.

YES   Indicates that the terminal has limited interactive capabilities.

If you omit the RIC parameter, the default value of NO is used.

3270 DEVICE DEFINITIONS

On a 3270 communication line, a device definition is different from a terminal definition. A 3270 terminal consists of one or more devices; terminal and device are not synonymous.

You must provide one TERMDEV or DEVICE statement for each device that accesses the communication line through the terminal you are defining. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal. Figures 6-14 and 6-15 present the formats of these statements for terminals on 3270 protocol lines.

For each terminal you define, you can declare up to 32 DEVICE statements. 3270 terminals can have up to 32 DEVICE statements for CDC-defined console devices or up to 7 DEVICE statements for CDC-defined line printer devices.

The following TERMINAL and DEVICE parameters provide device definitions for the terminal:

ABL	DBL	LK	TA
AUTOCON	DBZ	PG	UBL
B1	DI	PL	UBZ
B2	DT	PRI	
CN	HD	PW	
CT	HN	SDT	

For console devices:

device: DEVICE[,DT=CON,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,DBL=dwnblim,]

    [UBL=upblim,AUTOCON[=yn1],PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],]

    [LK=yn5,B1=b1,B2=b2,CN=cn,CT=ct,PG=pg,PL=pl,PW=pw].

For line printers:

device: DEVICE,DT=LP[,SDT=subdt,TA=trmaddr,DBZ=dwnlsiz,DBL=dwnblim,UBL=upblim,]

    [PRI[=yn1],DI[=yn2],PW=pw].

For site-defined devices:

device: DEVICE,DT=DT12[,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

    [DBL=dwnblim,UBL=upblim,AUTOCON[=yn1],PRI[=yn2],DI[=yn3],]

    [HN=node,HD[=yn4],LK=yn5,B1=b1,B2=b2,CN=cn,CT=ct,PG=pg,PL=pl,PW=pw].

device   The element name of the terminal device being defined. This name can be one through seven
            characters long; the first character must be a letter, the other characters can be letters
            or digits. If this statement is used within the set following a GROUP statement, the
            device name cannot be longer than five characters. This is the name used by the host or
            NPU operator to monitor and control the device. This name is required; there is no
            default value.

All other parameters are described in the text.

Figure 6-15. DEVICE Statement Format for Communication Lines of TIPTYPE=3270

The DT parameter is described first because it affects the validity, ranges, and defaults of the other parameters. The other parameters are described in alphabetical order.

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

If you specify a CDC-defined device type, the other device definition parameters on the DEVICE or TERMDEV statement:

> Are required by the NDL processor where indicated in the text
>
> Have the predefined default shown in the text
>
> Have the predefined ranges shown in the text
>
> Are checked for uniqueness and compatibility

If you specify a site-defined device type, the other device definition parameters on the DEVICE statement:

> Are not required by the NDL processor
>
> Have no predefined default
>
> Are not checked for uniqueness
>
> Are not checked for compatibility with other parameters on the same statement

When you specify the DT parameter, the following values are valid:

> CON   Identifies a CDC-defined console device.
>
> LP   Identifies a CDC-defined line printer device.
>
> DT12   Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.

If you omit the DT parameter, the value of CON is used.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). The ABL parameter has the following range of values:

$$1 \leq abl \leq 7$$

The value you declare should be greater than or equal to the downline block limit (DBL parameter value) of the terminal. Maximum line usage can be obtained by setting both DBZ and DBL so that 1 second of data is queued by CCP.

For console devices (DT of CON), the host queues abl - dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory but can reduce the number of times an application program is rolled out.

The default value for this parameter is 2 (DT of CON only).

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether CCP should automatically connect the console device and any associated batch devices to the selected host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

> NO   Indicates that CCP does not attempt automatic connection.
>
> YES   Indicates that CCP attempts automatic connection.

If you omit the AUTOCON parameter, the default value of NO is used for a DT of CON. If you specify this parameter without a value, the value of YES is used.

B1 PARAMETER

The B1 parameter specifies the character to be used as a user break 1 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 1 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 1 message as a job step interrupt from the terminal user.

This parameter is optional; it is allowed only for devices of DT=CON or DT=DT12. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    04, 0A, or 0D (characters EOT, LF, or CR)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

    Any value that you declare or use by default for the B2, CN, or CT parameters.

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B1 parameter or specify the value CCP, the default value is used (DT of CON only). The default value is:

TC Value	B1 Value Used	ASCII Character
3270	3A	:

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

B2 PARAMETER

The B2 parameter specifies the character to be used as a user break 2 indicator. When the terminal user enters this character as the only character on a line, CCP discards the block of data being transmitted to the terminal and sends a user break 2 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 2 message as a job step termination from the terminal user.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    04, 0A, or 0D (characters EOT, LF, or CR)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

    Any value that you declare or use by default for the B1, CN, or CT parameters.

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B2 parameter or specify the value CCP, the default value is used (DT of CON only). The default value is:

TC Value	B2 Value Used	ASCII Character
3270	29	)

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

CN PARAMETER

The CN parameter specifies the character to be used to abort (cancel) an input message. When the terminal user enters this character as the last character on a line, the network software discards the last message transmitted from the terminal.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

    00, 01, or 02 (characters NUL, SOH, or STX)

    04, 0A, or 0D (characters EOT, LF, or CR)

    3D, 7F, or 20 (characters =, DEL, or space)

    30 through 39 (characters 0 through 9)

    41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

    Any value that you declare or use by default for the B1, B2, or CT parameters.

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CN parameter or specify the value CCP, the default value is used (DT of CON only). The default value is:

TC Value	CN Value Used	ASCII Character
3270	28	(

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a termial definition command. When the terminal user enters this character as the first character on a line, CCP interprets the line as a command. Among other functions, CCP supports terminal definition commands to determine or change the values you have established for the following parameters:

B1	CT	PL
B2	HN	PW
CN	PG	TC

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

04, 0A, or 0D (characters EOT, LF, or CR)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

Any value that you declare or use by default for the B1, B2, or CN parameters.

Because the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, the default value is used (DT of CON only). The default value is:

TC Value	CT Value Used	ASCII Character
3270	25	%

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between the host computer and this terminal. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following value range:

$$1 \leq dbl \leq 7$$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the terminal. Maximum line usage can be obtained by setting both DBZ and DBL so that 1 second of data is queued by CCP.

For console devices (DT of CON), the host queues abl – dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

For batch devices, the NPU queues all downline blocks. No ABL value exists, so the DBL value alone determines the impact of block queuing on NPU and host resources. Large DBL values rapidly use up NPU memory; small DBL values may not keep the terminal busy.

If you omit the DBL parameter, a default value is used. Default values depend on the device type. The default values are:

DT Value	DBL Value Used
CON	2
LP	1

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. Larger DBZ values cause fewer disk accesses by the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional. The DBZ parameter has the following range of values:

$$1 \leq dbz \leq 2043$$

The value you declare should be chosen together with the value used for the DBL parameter. For batch devices (DT of LP), you should use a value that is a multiple of 640 characters (one physical record unit). If dbz is not a multiple of 640 for passive devices, NDL rounds the value you supply to the next even multiple, as follows:

Value Supplied	Value Used
$1 \leq dbz \leq 640$	640
$641 \leq dbz \leq 1280$	1280
$1281 \leq dbz \leq 2043$	1920

The NDL processor issues an informative message if it adjusts the DBZ value that you specify.

If you omit the DBZ parameter, a default value is used. Default values depend on the device type. The default values are:

DT Value	DBZ Value Used
CON	1280
LP	640

These defaults optimize use of the default value for the DBL parameter.

DI PARAMETER

The DI parameter is a stand-alone keyword that specifies a device as initially enabled or disabled. An enabled device is configured and serviced as soon as the communication line becomes active. A disabled device is neither configured nor serviced when the line becomes active.

If you initially disable the device, a host or NPU operator can change the device status to enabled. This change can be made only when the line becomes active (when a call is received on a dialup line, or when communications are established on a hardwired line).

This parameter is optional. If you specify the DI parameter, you must use one of the following values:

NO   Indicates that the device is initially enabled.

YES   Indicates that the device is initially disabled.

If you specify the DI parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

HD PARAMETER

The HD parameter controls whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional and is valid only for console devices.

If you specify this parameter, you must use one of the following values:

NO   Indicates that the full host availability display is not presented to the terminal user. The termial user receives only the host status message and the prompt message.

YES   Indicates that the host availability display is presented to the terminal user.

If you omit this parameter and only a single host is defined, the default value of NO is used. If you omit this parameter and multiple hosts are defined, the default value of YES is used.

The value of this parameter does not change when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can also change the value you specify.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device and any associated batch devices are connected to unless another path is selected by the terminal user.

The HN parameter is optional and is valid only for console devices. If you declare this parameter, you must use one of the following values:

NONE

Indicates that no node number is specified. This has the same effect as not declaring the parameter. A value of NONE cannot be declared if AUTOCON is also specified and there are logical links to more than one host terminating in the NPU being defined.

$1 \leq hn \leq 255$

Indicates the node number of the host that the console device and any associated batch devices are to be connected to.

The value declared for the HN parameter must be the same as the NODE value in a COUPLER statement within the same network definition. If all the COUPLER statements for all the logical links to the NPU being defined have equal HNAME parameters, the NDL processor provides a default host node. The default host node is the last coupler specified. If the HN parameter is not specified, the terminal user must select a host before a connection can be made. If the AUTOCON parameter is specified, then the HN parameter must also be specified.

The value of this parameter does not change when the terminal class is changed from the console. The terminal user can change the value you specify.

LK PARAMETER

The LK parameter specifies whether unsolicited messages from the NPU or host operator can appear at the terminal. This parameter is optional; it is allowed only for console devices and site-defined devices (DT of CON or DT of DT12).

If you declare this parameter, you must specify one of the following values:

CCP   Indicates that CCP uses the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that unsolicited messages are delivered to the terminal as soon as received by CCP.

YES   Indicates that unsolicited messages are discarded (locked out).

If you omit this parameter or specify the value CCP, the default value of NO is used (DT of CON only). The terminal user or the application program can change any value that you specify.

PG PARAMETER

The PG parameter specifies whether CCP waits at each output page boundary for terminal user acknowledgment before it displays the next page of data. In certain situations, page waiting can occur other than at page boundaries; CCP produces a prompting message (OVER..) when this type of page waiting occurs. The user's response to page waiting is entry of a line, usually empty.

CCP views a new page as beginning at the start of each downline message. CCP calculates the length of a page from the current values of the page width

and page length (see PW and PL Parameter descriptions). If the page width is infinite (PW=0), a page consists of one line less than the number of logical lines specified as the page length. If the page width is finite (PW is nonzero), a page consists of one line less than the number of physical lines specified by the page length; CCP calculates the number of physical lines by dividing each logical line into units less than or equal to the page width.

This parameter is optional; it is allowed only for console devices and site-defined devices (DT of CON or DT of DT12). If you specify this parameter, you must use one of the following reserved values:

CCP   Indicates that CCP uses the default setting appropriate for the terminal class; using this parameter is equivalent to omitting the parameter.

NO   Indicates that page waiting does not occur.

YES   Indicates that page waiting occurs.

When you specify PG=YES, you should also specify a nonzero value for the PL parameter. CCP cannot perform page waiting at the boundaries of infinitely long pages (PL=0).

If you omit the PG parameter or specify a value of CCP, the default value of YES is used.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program or the terminal user can change any value that you specify.

PL PARAMETER

The PL parameter specifies the number of physical lines per page of output for the device. If the device is configured for page waiting (PG parameter), any message containing more lines of output than the page length is interrupted by CCP for a page waiting response from the terminal user. The interruption occurs after line pl − 1 is output.

This parameter is optional; it is allowed for console devices, remote batch line printers, and site-defined devices only (DT of CON, DT of LP, or DT of DT12). If you specify the PL parameter, you must use one of the following values:

CCP

     Indicates that CCP uses the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$8 \le pl \le 255$

     Indicates the number of physical lines per page.

If you omit this parameter or specify a value of CCP, a default value is used (DT of CON only). The default value is 24. The default value for DT of LP is 64.

The value of this parameter reverts to the default value when the terminal class is changed from the console or changed by the application program. The terminal user or the application program can change any value that you specify.

PRI PARAMETER

The PRI parameter is a stand-alone keyword that indicates whether data to or from the device is to have traffic priority over that to or from other devices in the network. Data to or from a device that does not have traffic priority is suspended to relieve network congestion before data to or from terminals with traffic priority is suspended. Interactive devices should usually be given traffic priority over batch devices.

This parameter is optional. If you specify the PRI parameter, you must use one of the following values:

NO   Indicates that the device does not have data traffic priority.

YES   Indicates that the device has data traffic priority.

If you declare this parameter without a value, the value of YES is used. If you omit this parameter, the default value of NO is used.

PW PARAMETER

The PW parameter defines the number of characters per physical line of input or output for this device. This physical line length is also called the page width.

For console devices, output lines longer than pw characters are not divided but are counted as more than one line for page width calculations. If the PW value for the device is inappropriate, loss of visual fidelity can occur; the application program might divide the data into lines that are too short or too long for the screen's capacity, instead of allowing the terminal to wrap lines when needed.

This parameter is optional. If you specify the PW parameter, the value you can use depends on the device type. The following values are allowed for console devices (DT of CON):

CCP

     Indicates that CCP uses the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$20 \le pw \le 255$

     Indicates that the device can support physical lines no longer than the indicated decimal value.

The following values are allowed for line printer (DT of LP):

CCP

    Indicates that CCP uses the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$50 \leq pw \leq 255$

    Indicates that the device can support physical lines no longer than the indicated value.

For line printers, pw provides RBF with an indication of the character bytes that can comprise a physical record of output (the number of characters in a printer line).

If you omit this parameter or specify the value CCP, a default value is used. The default value depends on the device type. Default values are:

DT Value	PW Value Used
CON	80
LP	132

The value of this parameter for the console reverts to the default value when the terminal class is changed from the console or changed by the application program. The application program can change any value you specify for any device; the terminal user can change any value you specify for the console.

SDT PARAMETER

The SDT parameter specifies the subdevice type of the device you are defining. The subdevice type is the set of external characteristics of interest to the network software. This parameter is optional and valid only for devices with DT values of LP and DT12. The recognized reserved word values you can use and the defaults depend on the value you specify or use for the DT parameter.

If a DT value of LP is used, the following values are allowed definitions of a printer character set:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

CCP   Indicates that CCP uses the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12 Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13 Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14 Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15 Indicates a site-defined subdevice type that uses the subdevice type number 15.

If you omit the SDT parameter or specify the value CCP, the default value of A6 is used.

If a DT value of DT12 is used, the following values are allowed definitions of subdevice type:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

CCP   Indicates that CCP uses the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12 Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13 Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14 Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15 Indicates a site-defined subdevice type that uses the subdevice type number 15.

If you omit the SDT parameter, no default exists.

TA PARAMETER

The TA parameter defines the terminal address of the device within the terminal cluster. This parameter must match the code actually used to communicate with the device.

The TA parameter is required and can have the following range of values:

0 through 1F

    Indicates the hexadecimal terminal address of the terminal you are defining.

If you omit the TA parameter, the default value of 0 is used.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input messages and if you choose a UBL value smaller than this number, the application program becomes deadlocked (it waits for blocks of data that CCP is forced to discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following range of values:

$$1 \leq ubl \leq 31$$

If you omit the UBL parameter, a default value is used. The default value is 7.

The value you specify can be changed by the application program for consoles.

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline block can contain. CCP divides each message from the terminal into blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host.

You should choose a value that allows most of the logical lines entered by the device to fit into a single block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional; it is allowed only for console devices or site-defined devices (DT of CON or DT of DT12). The UBZ parameter has the following range of values for console devices (DT of CON):

$$0 \leq ubz \leq 2000$$

The range for site-defined devices (DT of DT12) is:

$$0 \leq ubz \leq 2043$$

If you specify 0 for a console device, CCP sends an upline block whenever it receives 100 characters.

The value you declare should be chosen together with the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes. For console devices (DT of CON), rounding occurs as follows:

Value Supplied	Value Used in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

The NDL processor issues an informative message if it adjusts the UBZ value that you specify.

If you omit the UBZ parameter, a default value is used. The default is 100 (DT of of CON only). The default optimizes use of the default value for the UBL parameter. The default value is the recommended value.

For the Message Control System (MCS), the upline block size must be set to 0.

The application program can change any value you specify. The terminal user can change the value within the following range:

$$0 \leq ubz \leq 200$$

BUFFERING OF DATA

It is desirable to maintain double buffering of data for each connection in the network. By using double buffering data will always be available to be sent over the communications line. Setting the ABL parameter to 2 and the DBL parameter to 1 will generally result in double buffering.

CCP buffer usage is optimized when downline block size follows the following relationship:

$$DBZ = 113 + n*118$$

where n is an integer and n + 1 is the number of buffers required in CCP to hold the block. This relationship is of most use for application programs that can control the size of the downline blocks they send to the network. Since it is necessary for a block to contain an integral number of logical lines, the lower bound of DBZ should not be smaller than the largest allowable PW for a device.

Default values for the ABL, DBL, and DBZ parameters are listed in table 6-2.

TABLE 6-2. ABL, DBL, AND DBZ DEFAULTS

Line Speed	ABL	DBL	DBZ
110	1	1	230
300	1	1	230
600	2	1	230
1200	2	1	230
2400	2	1	230
4800	2	1	460
9600	2	1	940

The parameters required and the values allowed on the GROUP, LINE, TERMINAL, TERMDEV, and DEVICE statements depend upon the line protocol used. This section describes the form of a statement applicable to site-defined protocol lines.

Control Data supplies Terminal Interface Programs (TIPs) for six communication protocols; the Network Definition Language provides support for additional site-written TIPs supporting other protocols. Such TIPs are identified by a TIPTYPE of TT12 through TT14 on LINE or GROUP statements in the network configuration file.

Because operating requirements and characteristics of site-written TIPs cannot be anticipated, NDL permits almost any combination of TERMINAL, TERMDEV, or DEVICE statement parameters for such TIPs. The rules for using these parameters are described in this section. Parameters are described in terms of their significance to CDC-written TIPs; these descriptions might not accurately portray the significance of a given parameter to site-written TIP.

RULES FOR DEFINITIONS

The NDL processor checks TERMINAL, DEVICE, or TERMDEV statement parameters for validity using the following rules:

Any parameter specified is legal.

No parameter is required.

Default values are not supplied for any parameters.

Any value declared for a parameter is legal, provided the value is within the range of values or reserved words that NDL recognizes for that keyword when the keyword is used for terminals or devices on lines of CDC-defined protocols.

Declared values for optional parameters are not checked for uniqueness where uniqueness is required for lines of CDC-defined protocols.

Declared values are not checked for compatibility where compatibility is required for lines of CDC-defined protocols.

LINE DEFINITION

You configure terminals on a site-defined protocol communication line with the following statements:

One LINE or GROUP statement that defines the line

One TERMINAL or TERMDEV statement for each terminal on the line

One DEVICE statement for each device accessing the line through a given terminal (a TERMDEV statement can be used if the terminal has only one device)

No dependencies exist among the parameter values declared on LINE, GROUP, TERMINAL, TERMDEV, and DEVICE statements except for those related to automatic recognition.

LINE STATEMENT PARAMETERS

Each LINE statement defines one communication line between a terminal and the NPU. There must be a LINE or GROUP statement defining each CLA port on the NPU that supports terminal access.

Figure 7-1 shows the format of the LINE statement and the valid parameter values for site-defined communication lines. Table 7-1 shows permitted line type values for the LTYPE parameter.

---

line: LINE,PORT=port,LTYPE=ltype,TIPTYPE=tiptype[,AUTO[=yn1],XAUTO[=yn2],ARSPEED=arspeed,DI[=yn3],]

    [IMDISC[=yn4],LCN=lcn,LSPEED=lspeed,DFL=dfl,FRAME=frame,RTIME=timer,RCOUNT=count,NSVC=svcirc,]

    [PSN=psn,DCE[=yn5],DTEA=locadr,AL=acclev,RC[=yn6],P90=fv90,...,P99=fv99].

line   The element name to be assigned to the communication line being defined. This name is
           required; there is no default value.

port   The hexadecimal number (1 ≤ port ≤ FE) of the port to which this communication line connects on
           the NPU currently being defined. Within an NDL program, the port number is independent of the
           number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE (254
           decimal). However, we strongly recommend that you assign port numbers consecutively, starting
           with 1. All values declared for PORT parameters must be unique within the current network
           definition of each NPU. The value specified for port cannot be the same as the number
           declared for a port in any other LINE or TRUNK statement (or within a GROUP statement
           expansion) for this NPU, and cannot be lower than the highest port number used by a trunk.
           This number is required; there is no default value.

Figure 7-1. Site-Defined Protocol LINE Statement Format (Sheet 1 of 3)

ltype   A reserved word value that identifies the type of communication line adapter/modem/circuit
          combination that is used on this line. This word must be supplied; there is no default value.
          The values allowed for this declaration are described in table 7-1.

tiptype   A reserved word value that identifies the type of Terminal Interface Program protocol required
          for the terminals on this line. This word is required; there is no default value. The values
          allowed for this declaration are:

                TT12   Site-written TIP that uses a TIP number of 12 for identification within the network
                        software is required.

                TT13   Site-written TIP that uses a TIP number of 13 for identification within the network
                        software is required.

                TT14   Site-written TIP that uses a TIP number of 14 for identification within the network
                        software is required.

yn1   An optional reserved word value (YES or NO) for an optional stand-alone keyword. This param-
          eter specifies whether this communication line is configured for automatic recognition of low
          line speeds (110 through 2400 bits per second) and/or recognition of protocol type by the TIP
          whenever a terminal is connected to the line. When AUTO or AUTO=YES is specified, the line is
          configured for automatic recognition. When automatic recognition is performed, the TIP attempts
          to determine as many definition parameters as possible at the time the line becomes active;
          recognized parameters are compared against parameters specified in the network definition file
          to identify the terminal accessing the line. If AUTO and XAUTO are omitted or if AUTO=NO or
          XAUTO=NO is specified, the TIP is not allowed to perform automatic recognition for the line;
          the line has a fixed configuration, and each TERMINAL or TERMDEV statement associated with this
          line must explicitly declare all addressing and protocol parameters legal for that terminal.

yn2   An optional reserved word value (YES or NO) for an optional stand-alone keyword. This param-
          eter specifies whether this communication line is configured for automatic recognition of high
          line speeds (600 through 9600 bits per second) and/or recognition of protocol type by the TIP
          whenever a terminal is connected to the line. When XAUTO or XAUTO=YES is specified, the line
          is configured for automatic recognition. When automatic recognition is performed, the TIP
          attempts to determine as many definition parameters as possible at the time the line becomes
          active; recognized parameters are compared against parameters specified in the network defini-
          tion file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or if
          AUTO=NO or XAUTO=NO is specified, the TIP is not allowed to perform automatic recognition for
          the line; the line has a fixed configuration, and each TERMINAL and DEVICE or TERMDEV statement
          associated with this line must explicitly declare all addressing and protocol parameters legal
          for that terminal.

arspeed   An optional reserved word value (YES or NO) that indicates whether the user can change the line
          speed by using the AR TIP command.

yn3   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter
          specifies the status assigned to this communication line at network initiation. When DI or
          DI=YES is specified, the line is disabled at network initiation and cannot be used until the
          HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is given an initial
          status of enabled unless the NOP or HOP specifies otherwise.

yn4   An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect
          feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on the
          line is disconnected immediately after the terminal has logged out (disconnected) from the
          host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is disconnected
          only after a two minute timer expires.

lcn   The logical channel number ($0 \le lcn \le 255$) which corresponds to the lowest numbered channel
          that a DTE CCP can use to make an outgoing application-to-application call. If CCP is a DCE,
          it corresponds to the highest numbered virtual circuit CCP can use to make outgoing
          application-to-application calls. When lcn is not specified, no default exists.

lspeed   The baud rate used by the modems or devices accessing this line. This parameter can only be
          used to configure asynchronous lines (LTYPE of A1, A2, or A6). When the line is configured for
          automatic recognition (AUTO is declared), this parameter cannot be used. This parameter is
          optional when an asynchronous line is not configured for automatic recognition of terminals;
          the following values are recognized:

                110   134   150   300   600   1200   2400   4800   9600   19200   38400

          When lspeed is not declared for an asynchronous line configured without automatic recognition,
          no default exists.

Figure 7-1. Site-Defined Protocol LINE Statement Format (Sheet 2 of 3)

dfl   The maximum decimal number (16 $\leq$ dfl $\leq$ 1024) of 8-bit characters or octets in an X.25 data
        packet.

frame   The frame window (1 (1 $\leq$ frame $\leq$ 7); the maximum number of outstanding unacknowledged packets
        between the network processing unit and the X.25 network (this is the K parameter defined by
        the X.25 protocol standard).

timer   The decimal retransmission timing period (1 $\leq$ timer $\leq$ 25500) in milliseconds; the time that
        must elapse before retransmission of an unackowledged X.25 data frame is attempted (this is the
        T1 parameter defined by the X.25 protocol standard). NDLP rounds the value specified for timer
        to the next multiple of 100.

count   The decimal retransmission count (1 $\leq$ count $\leq$ 15); the number of times retransmission of an
        unacknowledged X.25 data frame is attempted (this is the N2 parameter defined by the X.25
        protocol standard).

svcirc The decimal number (0 $\leq$ svcirc $\leq$ 255) of available switched virtual circuits (SVC) for this
        line. The parameter must be identical to the number of switched virtual circuits contracted
        from the packet switching network and must conform to the following rules:

            The total number of virtual circuits defined (by the NCIR parameter on the TERMINAL or
            TERMDEV statement) for each STIP must be $\leq$ svcirc.

            The total number of virtual circuit definitions must be $\leq$ 255; svcirc + number of permanent
            virtual circuit (PVC) definitions $\leq$ 255.

psn   A reserved word value identifying the packet switching network to which the line is connected.
        The following values are recognized:

            CDSN   Packet switching network is compatible with the commercial CDSN network.

            C120   Packet switching network is compatible with the CYBER 120.

            DATAPAC Packet switching network is compatible with the commercial DATAPAC network.

            PSN253   Site-defined packet switching network type, identified by the number 253 within
                     the network software.

            PSN254   Site-defined packet switching network type, identified by the number 254 within
                     the network software.

            PSN255   Site-defined packet switching network type, identified by the number 255 within
                     the network software.

            TELENET Packet switching network is compatible with the commercial TELENET network.

            TRNSPAC Packet switching network is compatible with the commercial TRANSPAC network.

            TYMNET   Packet switching network is compatible with the commercial TYMNET network.

            UNINET   Packet switching network is compatible with the commercial UNINET network.

yn5   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter
        tells the TIP what role to play in an X.25 protocol. If DCE or DCE=YES is specified, the TIP
        must act as data circuit terminating equipment (DCE). If DCE is omitted or DCE=NO is speci-
        fied, the TIP must act as data terminal equipment (DTE) protocol.

locadr The decimal address (0 $\leq$ locadr $\leq$ 99) assigned to the local end of the X.25 link. This value
        specifies the address by which the X.25 protocol identifies calls to and from the connected
        NPU. The value declared should be the calling DTE address for outgoing call requests when
        DCE=YES is specified.

acclev The decimal access level limit (0 $\leq$ acclev $\leq$ 7) for the line. The lowest access level is 0
        (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level
        limit is associated with the line (equivalent to specifying AL=0).

yn6   An optional reserved word value (YES or NO) which specifies whether the reconfiguration
        indicator is enabled for this line. When RC or RC=YES is specified, the terminal character-
        istics are reset to their original NDL values (or to a default if no NDL values were specified)
        when the terminal disconnects from a host. When RC is omitted or RC=NO is specified, the
        reconfiguration indicator is not enabled for this line.

fv9i   The hexadecimal field value (0 $\leq$ fv9i $\leq$ FF) to use for the corresponding field number within
        the Terminal Interface Program configuration information transmitted for this line.

Figure 7-1. Site-Defined Protocol LINE Statement Format (Sheet 3 of 3)

TABLE 7-1. SITE-DEFINED LINE TYPE DEFINITIONS

LTYPE Value	Transmission Mode	Transmission Operation	Circuit Type	Modem Type†	CLA Type	Maximum Speed, Bits per Second	Carrier Type
S1	Synchronous	Half-duplex	Switched (Dialup)	RS232C, Bell 201A/208B compatible	2560-1	4800	Controlled
S2	Synchronous	Full-duplex but operating half-duplex	Dedicated (Hardwired)	RS232C, Bell 201B/208A compatible Bell 301/303 compatible V.35 Standard compatible	2560-1 2560-2 2560-3	19200 50000 (HASP only) 56000 (HASP only)	Controlled
S3	Synchronous	Full-duplex	Dedicated (Hardwired)	RS232C, Bell 201B/208A compatible Bell 301/303 compatible V.35 Standard compatible	2560-1 2560-2 2560-3	19200 50000 (HASP only) 56000 (HASP only)	Constant
S4 (For site-defined use)	Synchronous	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned
A1	Asynchronous	Full-duplex	Switched (Dialup)	RS232C, Bell 103E/113/212A compatible	2561-1	9600	Constant
A2	Asynchronous	Full-duplex	Dedicated (Hardwired)	RS232C, Bell 103E/113/212A compatible	2561-1	9600	Constant
A6 (For site-defined use)	Asynchronous	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned
H1	Bit-oriented synchronous HDLC	Full-duplex	Dedicated (Hardwired)	RS232C, Bell 201B compatible V.35 Standard compatible	2563-1 2563-2	19200 56000	Constant
H2 (For site-defined use)	HDLC	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned

†Modem types supported by each type of CLA may differ.

You use the AUTO or XAUTO parameter to define a communication line as having a fixed or an automatic recognition configuration. You can use either of these configuration types for a dialup line or for a dedicated (hardwired) line.

If you define a fixed-configuration line by omitting AUTO and XAUTO or by specifying AUTO=NO or XAUTO=NO, you must know and specify certain characteristics of the terminals and devices that will use it. Only terminals and devices with the specified characteristics can use that line.

If you define an automatic recognition line by specifying AUTO, XAUTO, AUTO=YES, or XAUTO=YES, your site-written TIP determines these characteristics when a terminal becomes active on the line. This allows more flexible access to the network.

A terminal on an automatic recognition line is not completely configured until it becomes active. At that time, the Communications Supervisor (CS) compares the determined characteristics against the characteristics you have defined for each terminal configured on the line. CS uses the first terminal definition with the closest match to finish configuring the terminal. Your site-written TIP then services the terminal devices according to the finished configuration.

Only a terminal that completely matches all declared values can use the line. It must operate with the characteristics you declare for any required or optional parameters in your terminal definition, or the terminal user must change those characteristics to match. If you declare AUTOREC instead of values for all of the automatically recognized parameters, then any terminal will match the terminal definition.

If you declare values instead of AUTOREC for some of the automatically recognized parameters, then any terminal that successfully accesses the line and has characteristics matching the declared values will match your terminal definition. If you declare values instead of AUTOREC for all of the automatically recognized parameters, then only a terminal with automatically recognized characteristics that are the same as your values will match the terminal definition.

You can specify more than one terminal definition on an automatic recognition line. Each definition should vary from the others in one of the automatically recognized characteristics; unless differences exist, CS will not use any definition other than the first one.

Using automatic recognition increases the number of terminal devices that potentially can use the line. You can configure a switchable line for automatic recognition with more logical terminal devices than the physical terminal devices that simultaneously can access it; only one physical device can access the line at a time.

For example, the statement

    LN3F: LINE PORT=0A,TIPTYPE=TT12,LTYPE=S4,AUTO.

defines a switchable (dialup), synchronous line (not in the SVLnet example, figure 2-1 in section 2) for automatic recognition of any synchronous protocol terminals. The line is identified as LN3F, connects to the NPU at port 0A, and is enabled by default at network initiation.

If the fixed-configuration form of the statement

    LN3F: LINE PORT=0A,LTYPE=S4,TIPTYPE=TT12.

were used instead, all of the terminals capable of accessing this switchable line would have to be completely and explicitly configured on their TERMINAL and DEVICE statements.

As another example of a fixed-configuration line, the statement

    LN2E: LINE PORT=0A,LTYPE=S4,TIPTYPE=TT14.

defines a hardwired (dedicated), enabled, synchronous line for multileaving workstation terminals, accessing the NPU (also not shown in SVLnet) through port 0A.

The following LINE statements are not equivalent:

    LN1A: LINE PORT=9,LTYPE=A6,TIPTYPE=TT13,
          LSPEED=300.

    LN1A: LINE PORT=9,LTYPE=A6,TIPTYPE=TT13.

Both statements define a switchable, enabled, fixed-configuration asynchronous line accessing the NPU (not shown in SVLnet) through port 9. The first line is configured to support modems at 300 baud; the second line has an unknown baud rate, probably determined by the hardware. All terminals accessing this line must be explicitly configured because the line is not configured for automatic recognition.

GROUP STATEMENT PARAMETERS

You can use the GROUP statement in place of the LINE statement when you want to repeat a line definition a specified number of times.

NDLP creates identical line, terminal, and device definitions (except for port number, line name, and device name) the number of times specified by the NI parameter. NDLP generates unique port numbers by incrementing the previously defined port number by one. NDLP generates unique line and device element names by adding the LINE's port number in two-digit hexadecimal form to the one- to five-character root element name that you specify.

Figure 7-2 shows the format of the GROUP statement and the valid parameter values for site-defined communication lines. Table 7-1 shows permitted line type values for the LTYPE parameter.

```
group:   GROUP,PORT=port,LTYPE=ltype,TIPTYPE=tiptype[,AUTO[=yn1],XAUTO[=yn2],ARSPEED=arspeed,DI[=yn3],]

   [IMDISC[=yn4],LCN=lcn,LSPEED=lspeed,DFL=dfl,FRAME=frame,RTIME=timer,RCOUNT=count,NSVC=svcirc,]

   [PSN=psn,DCE[=yn5],DTEA=locadr,AL=acclev,RC[=yn6],P90=fv90,...,P99=fv99,NI=iter].
```

group   The root name to use for generating the element names assigned to the communication lines and
           devices being defined. This name, which cannot be longer than five characters, must result in
           element names that are unique within the network division currently being described. This
           name is required; there is no default value.

port   The hexadecimal number (1 $\leq$ port $\leq$ FE) of the port to which the first of these communication
           lines connects on the NPU currently being defined. Within an NDL program, the port number is
           independent of the number of ports on the NPU; for example, a 128-port NPU can have a port
           numbered FE (254 decimal). However, we strongly recommend that you assign port numbers con-
           secutively, starting with 1. All values declared for PORT parameters must be unique within
           the current network definition of each NPU. The value specified for port cannot be the same
           as the number declared for a port in any LINE or TRUNK statement (or within any other GROUP
           statement expansion) for this NPU, and cannot be lower than the highest port number used by a
           trunk. This number is required; there is no default value.

ltype   A reserved word value that identifies the type of communication line adapter/modem/circuit
           combination that is used on this line. This word must be supplied; there is no default value.
           The values allowed for this declaration are described in table 7-1.

tiptype   A reserved word value that identifies the type of Terminal Interface Program protocol required
           for the terminals on this line. This word is required; there is no default value. The values
           allowed for this declaration are:

           TT12   Site-written TIP that uses a TIP number of 12 for identification within the network
                  software is required.

           TT13   Site-written TIP that uses a TIP number of 13 for identification within the network
                  software is required.

           TT14   Site-written TIP that uses a TIP number of 14 for identification within the network
                  software is required.

yn1   An optional reserved word value (YES or NO) for an optional stand-alone keyword. This
           parameter specifies whether this communication line is configured for automatic recognition of
           low line speeds (110 through 2400 bits per second) and/or recognition of protocol type by the
           TIP whenever a terminal is connected to the line. When AUTO or AUTO=YES is specified, the
           line is configured for automatic recognition. When automatic recognition is performed, the
           TIP attempts to determine as many definition parameters as possible at the time the line
           becomes active; recognized parameters are compared against parameters specified in the network
           definition file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or
           if AUTO=NO or XAUTO=NO is specified, the TIP is not allowed to perform automatic recognition
           for the line; the line has a fixed configuration, and each TERMINAL or TERMDEV statement
           associated with this line must explicitly declare all addressing and protocol parameters legal
           for that terminal.

yn2   An optional reserved word value (YES or NO) for an optional stand-alone keyword. This
           parameter specifies whether this communication line is configured for automatic recognition of
           high line speeds (600 through 9600 bits per second) and/or recognition of protocol type by the
           TIP whenever a terminal is connected to the line. When XAUTO or XAUTO=YES is specified, the
           line is configured for automatic recognition. When automatic recognition is performed, the
           TIP attempts to determine as many definition parameters as possible at the time the line
           becomes active; recognized parameters are compared against parameters specified in the network
           definition file to identify the terminal accessing the line. If AUTO and XAUTO are omitted or
           if AUTO=NO or XAUTO=NO is specified, the TIP is not allowed to perform automatic recognition
           for the line; the line has a fixed configuration, and each TERMINAL and DEVICE or TERMDEV
           statement associated with this line must explicitly declare all addressing and protocol
           parameters legal for that terminal.

arspeed   An optional reserved word value (YES or NO) that indicates whether the user can change the
           line speed by using the AR TIP command.

yn3   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this
           parameter specifies the status assigned to this communication line at network initiation.
           When DI or DI=YES is specified, the line is disabled at network initiation and cannot be used
           until the HOP or NOP enables it. If DI is omitted or DI=NO is specified, the line is given an
           initial status of enabled unless the NOP or HOP specifies otherwise.

                    Figure 7-2. Site-Defined Protocol GROUP Statement Format (Sheet 1 of 3)

yn4   An optional reserved word value (YES or NO) which specifies whether the Immediate Disconnect
        feature is enabled for this line. When IMDISC or IMDISC=YES is specified, any terminal on the
        line is disconnected immediately after the terminal has logged out (disconnected) from the
        host. When IMDISC is omitted or IMDISC=NO is specified, a terminal on the line is
        disconnected only after a two minute timer expires.

lcn   The logical channel number ($0 \leq lcn \leq 255$) which corresponds to the lowest numbered channel
        that a DTE CCP can use to make an outgoing application-to-application call. If CCP is a DCE,
        it corresponds to the highest numbered virtual circuit CCP can use to make outgoing
        application-to-application calls. When lcn is not specified, no default exists.

lspeed The baud rate used by the modems or devices accessing this line. This parameter can only be
        used to configure asynchronous lines (LTYPE of A1, A2, or A6). When the line is configured
        for automatic recognition (AUTO is declared), this parameter cannot be used. This parameter
        is optional when an asynchronous line is not configured for automatic recognition of
        terminals; the following values are allowed:

             110   134   150   300   600   1200   2400   4800   9600   19200   38400

        When lspeed is not declared for an asynchronous line configured without automatic
        recognition, no default exists.

dfl   The maximum decimal number ($16 \leq dfl \leq 1024$) of 8-bit characters or octets in an X.25 data
        packet.

frame   The frame window ($1 \leq frame \leq 7$); the maximum number of outstanding unacknowledged packets
        between the network processing unit and the X.25 network (this is the K parameter defined by
        the X.25 protocol standard).

timer   The decimal retransmission timing period ($1 \leq timer \leq 25500$) in milliseconds; the time that
        must elapse before retransmission of an unackowledged X.25 data frame is attempted (this is
        the T1 parameter defined by the X.25 protocol standard). NDLP rounds the value specified for
        timer to the next multiple of 100.

count   The decimal retransmission count ($1 \leq count \leq 15$); the number of times retransmission of an
        unacknowledged X.25 data frame is attempted (this is the N2 parameter defined by the X.25
        protocol standard).

svcirc The decimal number ($0 \leq svcirc \leq 255$) of available switched virtual circuits (SVC) for this
        line. The parameter must be identical to the number of switched virtual circuits contracted
        from the packet switching network and must conform to the following rules:

             The total number of virtual circuits defined (by the NCIR parameter on the TERMINAL or
             TERMDEV statement) for each STIP must be $\leq$ svcirc.

             The total number of virtual circuit definitions must be $\leq$ 255; svcirc + number of permanent
             virtual circuit (PVC) definitions $\leq$ 255.

psn   A reserved word value identifying the packet switching network to which the line is
        connected. The following values are recognized:

        CDSN   Packet switching network is compatible with the commercial CDSN network.

        DATAPAC   Packet switching network is compatible with the commercial DATAPAC network.

        PSN253   Site-defined packet switching network type, identified by the number 253 within
                    the network software.

        PSN254   Site-defined packet switching network type, identified by the number 254 within
                    the network software.

        PSN255   Site-defined packet switching network type, identified by the number 255 within
                    the network software.

        TELENET   Packet switching network is compatible with the commercial TELENET network.

        TRNSPAC   Packet switching network is compatible with the commercial TRANSPAC network.

Figure 7-2. Site-Defined Protocol GROUP Statement Format (Sheet 2 of 3)

TYMNET	Packet switching network is compatible with the commercial TYMNET network.
UNINET	Packet switching network is compatible with the commercial UNINET network.
yn5	An optional reserved word value (YES or NO) for an optional stand-alone keyword; this parameter tells the TIP what role to play in an X.25 protocol. If DCE or DCE=YES is specified, the TIP must act as data circuit terminating equipment (DCE). If DCE is omitted or DCE=NO is specified, the TIP must act as data terminal equipment (DTE) protocol.
locadr	The decimal address (0 $\leq$ locadr $\leq$ 99) assigned to the local end of the X.25 link. This value specifies the address by which the X.25 protocol identifies calls to and from the connected NPU. The value declared should be the calling DTE address for outgoing call requests when DCE=YES is specified.
acclev	The decimal access level limit (0 $\leq$ acclev $\leq$ 7) for the line. The lowest access level is 0 (unclassified), the highest is 7. If AL is omitted or AL=NONE is specified, no access level limit is associated with the line (equivalent to specifying AL=0).
yn6	An optional reserved word value (YES or NO) which specifies whether the reconfiguration indicator is enabled for this line. When RC or RC=YES is specified, the terminal characteristics are reset to their original NDL values (or to a default if no NDL values were specified) when the terminal disconnects from a host. When RC is omitted or RC=NO is specified, the reconfiguration indicator is not enabled for this line.
fv9i	The hexadecimal field value (0 $\leq$ fv9i $\leq$ FF) to use for the corresponding field number within the Terminal Interface Program configuration information transmitted for this line.
iter	The decimal number (1 $\leq$ iter $\leq$ 254) of iterations of this line definition and of all TERMINAL, TERMDEV, or DEVICE statements following this GROUP statement. This parameter is optional; the default is 1.

Figure 7-2. Site-Defined Protocol GROUP Statement Format (Sheet 3 of 3)

Using the GROUP statement can reduce the number of definitions you must provide in your NDL program without reducing the number of definitions provided in the network configuration file. An example of GROUP statement use is shown in figure 7-3.

You use the AUTO or XAUTO parameter to define a communication line as having a fixed or an automatic recognition configuration. You can use either of these configuration types for a dialup line or for a dedicated (hardwired) line.

If you define a fixed-configuration line by omitting AUTO and XAUTO or by specifying AUTO=NO or XAUTO=NO, you must know and specify certain characteristics of the line, and of the terminals and devices that will use it. Only terminals and devices with the specified characteristics can use that line.

If you define an automatic recognition line by specifying AUTO, XAUTO, XAUTO=YES, or AUTO=YES, your site-written TIP determines these characteristics when a terminal becomes active on the line. This allows more flexible access to the network.

A terminal on an automatic recognition line is not completely configured until it becomes active. At that time, the Communications Supervisor (CS) compares the determined characteristics against the characteristics you have defined for each terminal configured on the line. CS uses the first terminal definition with the closest match to finish configuring the terminal. Your site-written TIP then services the terminal devices according to the finished configuration.

Only a terminal that completely matches all declared values can use the line. It must operate with the characteristics you declare for any required or optional parameters in your terminal definition, or the terminal user must change the characteristics to match. If you declare AUTOREC instead of values for all of the automatically recognized parameters, then any terminal will match the terminal definition.

If you declare values instead of AUTOREC for some of the automatically recognized parameters, then any terminal that successfully accesses the line and has characteristics matching the declared values will match your terminal definition. If you declare values instead of AUTOREC for all of the automatically recognized parameters, then only a terminal with automatically recognized characteristics that are the same as your values will match the terminal definition.

You can specify more than one terminal definition on an automatic recognition line. Each definition should vary from the others in one of the automatically recognized characteristics. Unless differences exist, CS will not use any definition other than the first one.

Using automatic recognition increases the number of terminal devices that potentially can use the line. You can configure a switchable line for automatic recognition with more logical terminal devices than the physical terminal devices that simultaneously can access it; only one physical device can access the line at a time.

```
The following set of statements:

    LN3F: GROUP,PORT=OA,LTYPE=S4,
        TIPTYPE=TT12,NI=2.
        TERMINAL,STIP=M4A,TC=200UT,CA=70.
            DEVD: DEVICE,DT=CON,TA=60.
            DEVL: DEVICE,DT=LP,TA=60.
            DEVC: DEVICE,DT=CR,TA=60.

    LN1A: GROUP,PORT=9,LTYPE=A6,
        TIPTYPE=TT13,LSPEED=300,NI=5.
            DEVA: TERMDEV,TC=713,PRI.


is interpreted by the NDL processor as if it had
been written:

    LN3FOA: LINE,PORT=OA,LTYPE=S4,
        TIPTYPE=TT12.
        TERMINAL,STIP=M4A,TC=200UT,CA=70.
            DEVDOA: DEVICE,DT=CON,TA=60.
            DEVLOA: DEVICE,DT=LP,TA=60.
            DEVCOA: DEVICE,DT=CR,TA=60.

    LN3FOB: LINE,PORT=OB,LTYPE=S4,
        TIPTYPE=TT12.
        TERMINAL,STIP=M4A,TC=200UT,CA=70.
            DEVDOB: DEVICE,DT=CON,TA=60.
            DEVLOB: DEVICE,DT=LP,TA=60.
            DEVCOB: DEVICE,DT=CR,TA=60.

    LN1A09: LINE,PORT=9,LTYPE=A6,
        TIPTYPE=TT13,LSPEED=300.
            DEVA09: TERMDEV,TC=713,PRI=YES.

    LN1AOA: LINE,PORT=OA,LTYPE=A6,
        TIPTYPE=TT13,LSPEED=300.
            DEVAOA: TERMDEV,TC=713,PRI=YES.

    LN1AOB: LINE,PORT=OB,LTYPE=A6,
        TIPTYPE=TT13,LSPEED=300.
            DEVAOB: TERMDEV,TC=713,PRI=YES.

    LN1AOC: LINE,PORT=OC,LTYPE=A6,
        TIPTYPE=TT13,LSPEED=300.
            DEVAOC: TERMDEV,TC=713,PRI=YES.

    LN1AOD: LINE,PORT=OD,LTYPE=A6,
        TIPTYPE=TT13,LSPEED=300.
            DEVAOD: TERMDEV,TC=713,PRI=YES.
```

Figure 7-3. Site-Defined Protocol
GROUP Statement Expansion

For example, the statement

    LN1A: GROUP,PORT=9,TIPTYPE=TT13,LTYPE=A6,
        AUTO,NI=2.

defines two switchable (dialup), asynchronous lines
(not shown in figure 2-1 of section 2) for automatic
recognition of asynchronous protocol terminals.
The lines are identified as LN1A09 and LN1AOA, con-
nect to the NPU at ports 9 and OA, and are enabled
by default at network initiation.

If the fixed-configuration form of the statement

    LN1A: GROUP,PORT=9,LTYPE=A6,TIPTYPE=TT13,NI=2.

were used instead, all of the terminals capable of
accessing these switchable lines would have to be
completely and explicitly configured on the TERMINAL
and DEVICE statements following the GROUP statement.

TERMINAL DEFINITIONS

You can declare up to 255 TERMINAL or TERMDEV
statements for a communication line. You must pro-
vide one TERMINAL or TERMDEV statement for each
terminal that can access the line. If you use a
TERMDEV statement, you cannot use a DEVICE statement
for the same terminal. Figures 7-4 and 7-5 present
the formats of these statements for terminals on
lines supported by site-written TIPs.

```
TERMINAL  [,STIP=stiptyp,TC=trmclas,]

    [CSET=charset,TSPEED=trmsped,]

    [CA=clsadr,RIC=yn1,]

    [CO=conord,BCF[=yn2],MREC=rec,]

    [W=pacwndw,NCIR=numcir,PAD=string,]

    [NEN=encir,COLLECT[=yn3],EOF[=yn4]].


Parameters are described in the text.
```

Figure 7-4. TERMINAL Statement Format for
Site-Defined Communication Line Protocols

The following TERMINAL and TERMDEV parameters pro-
vide the terminal definition for the line:

BCF	NEN
CA	PAD
CO	RIC
COLLECT	STRIP
CSET	TC
EOF	TSPEED
MREC	W
NCIR	

The STIP and TC parameters are described first
because they are described in that order in sections
4 through 6. The other parameters are described in
alphabetical order.

STIP PARAMETER

The STIP parameter specifies the subTIPtype required
to support the terminal definition. This parameter
is optional. When you declare a value for STIP, it
must be one of the following reserved words:

AUTOREC   Specifies that the subTIPtype is to be
          determined by your site-written TIP
          when automatic recognition of the
          terminal occurs (valid only if TC is
          specified); using this value is
          equivalent to omitting the parameter.

M4A   Identifies a synchronous terminal that
          uses CDC Mode 4A protocol.

M4C   Identifies a synchronous terminal that
          uses CDC Mode 4C protocol.

N2741   Identifies an asynchronous terminal
          that does not use IBM 2741 protocol.

```
device:  TERMDEV[,STIP=stiptyp,TC=trmclas,CSET=charset,TSPEED=trmsped,CA=clsadr,RIC=yn1,]

    [CO=conord,BCF[=yn2],MREC=rec,W=pacwndw,NCIR=numcir,PAD=string,NEN=encir,]

    [COLLECT[=yn3],DT=devtyp,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

    [DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,DO=devord,STREAM=streamno,AUTOCON[=yn4],]

    [PRI[=yn5],DI[=yn6],HN=node,HD[=yn7],LK=yn8,AB=ab,BR=br,CP=cp,BS=bs,B1=b1,B2=b2,CI=ci,]

    [CN=cn,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,EBO=ebo,ELX=elx,ELR=elr,]

    [ELO=elo,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,IN=in,LI=li,OP=op,OC=oc,]

    [PA=pa,PG=pg,PL=pl,PW=pw,EOF[=yn10],RTS[=yn11],MCI=mc,MLI=ml,P90=fv90,...,P99=fv99].


device      The element name of the terminal device being defined.  This name can be one through seven
            characters long.  The first character must be a letter; the other characters can be
            letters or digits.  If this statement is used within the set following a GROUP statement
            or as a switched virtual circuit definition, the device name cannot be longer than five
            characters.  This is the name used by the host or NPU operator to monitor and control the
            device.  This name is required; there is no default value.

All other parameters are described in the text.
```

Figure 7-5. TERMDEV Statement Format for Site-Defined Communication Line Protocols

PAD   Identifies a terminal that is con-
          nected via the asynchronous stop-start
          (X.29) packet assembly/disassembly
          protocol.

POST   Identifies a synchronous terminal that
          supports only postprinting format con-
          trol and IBM HASP protocol.

PRE   Identifies a synchronous terminal that
          supports preprinting format control
          and IBM HASP protocol.

USER   Identifies an asynchronous terminal
          using a site-defined form of X.25
          packet protocol.

XAA   Identifies the application-to-
          application X.25 subTIP.

2741   Identifies an asynchronous terminal
          that uses IBM 2741 protocol.

2780   Identifies a bisynchronous terminal
          that uses IBM 2780 protocol.

3780   Identifies a bisynchronous terminal
          that uses IBM 3780 protocol.

If you omit the STIP parameter, the default is
unknown and possibly undefined.

TC PARAMETER

The TC parameter specifies the terminal class
appropriate for the device. Supported devices are
grouped into terminal classes, according to their
hardware characteristics.

No default values exist for the characteristics
associated with an archetype terminal on a com-
munication line with a site-defined protocol. Each
CDC-defined terminal class has a range of possible

values for the characteristics associated with an
archetype terminal. These ranges determine the
values available for you to specify on the TERMDEV
or DEVICE statements. A site-defined terminal class
has no predefined range; any value recognized for
one of the CDC-defined terminal classes is valid
for any site-defined terminal class.

The mnemonics identifying the terminal classes are
derived from the identification of the archetype
terminals defining the classes, with the following
exceptions:

    HASP and HPRE each identify the communication
    protocol used by a terminal as the archetype.

    You can use TC28, TC29, TC30, and TC31 to
    identify site-defined terminal classes, cor-
    responding to reserved Network Access Method
    (NAM) terminal class numbers.

The TC parameter is optional. When you declare a
value for TC, it must be one of the following
reserved words:

CCP   Specifies that CCP is to provide the
          terminal class appropriate for the sub-
          TIPtype; using this value is equivalent
          to omitting the parameter.

HASP   Identifies a terminal or device com-
          patible with the postprint variant of
          IBM HASP protocol.

HPRE   Identifies a terminal or device com-
          patible with the preprint variant of
          IBM HASP protocol.

H2000   Identifies a terminal device compatible
          with Hazeltine 2000 series equipment
          emulating a teletypewriter.

M33   Identifies a terminal device compatible
          with Teletype Model 30 series equipment.

M40   Identifies a terminal device compatible with Teletype Model 40-2 equipment.

TC28   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 28.

TC29   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 29.

TC30   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 30.

TC31   Identifies a site-defined archetype terminal that uses the NAM internal terminal class number of 31.

T4014   Identifies a terminal device compatible with Tektronix 4014 series equipment emulating a teletypewriter.

X364   Identifies a terminal device compatible with the ANSI X3.64 standard (DEC VT100 or CDC 722-30).

200UT   Identifies a terminal or device compatible with the CDC 200 User Terminal.

711   Identifies a terminal or device compatible with the CDC 711 series terminal that uses the 4C variant of the Mode 4 protocol.

713   Identifies a terminal device compatible with CDC Model 713, 722-10, 751, 752, or 756 series equipment.

714   Identifies a terminal or device compatible with the CDC 714-10/20 series terminal that uses the 4C variant of the Mode 4 protocol.

714X   Identifies a terminal or device compatible with the CDC 714-30 series terminal that uses the 4C variant of the Mode 4 protocol.

721   Identifies a terminal device compatible with CDC Model 721 equipment.

734   Identifies a terminal or device compatible with the CDC 734 series terminal that uses the 4A variant of the Mode 4 protocol.

2741   Identifies a terminal device compatible with IBM 2741 equipment.

2780   Identifies a bisynchronous terminal using the IBM 2780 protocol.

3270   Identifies a terminal that is compatible with the IBM 3270.

3780   Identifies a bisynchronous terminal using the IBM 3780 protocol.

If the TC parameter is specified with a value of TC28, TC29, TC30, or TC31, then CCP must also be modified to accept these values. If the appropriate support code for TC28, TC29, TC30, or TC31 is not added to CCP, these values will be considered invalid by CCP.

If you omit the TC parameter, the default is unknown and possibly undefined.

BCF PARAMETER

The BCF parameter indicates whether the terminal supports compression of blanks in upline and downline data. Blank compression consists of deleting blanks that occur in multiples and replacing each multiple with a code set sequence that represents the number deleted.

When the network software compresses blanks in downline data, it does so for all devices within the terminal. When the network software receives a compressed blank sequence in upline data from any device within the terminal, it expands the data by replacing each sequence with the appropriate number of blanks.

This stand-alone parameter is optional. If you specify this parameter, you can use one of the following values:

NO   Indicates that the terminal does not compress blanks in its input and cannot process compressed blanks received as output.

YES   Indicates that the terminal compresses blanks in its input and can process compressed blanks received as output.

If you specify this parameter without a value, the value of YES is used. If you omit this parameter, the default is unknown and possibly undefined.

CA PARAMETER

The CA parameter specifies the hardware cluster address for all the devices accessing the line through one terminal. There is no default value for a cluster address.

If you supply this parameter for an automatic recognition line, the value need not be unique for each terminal on the line. If you supply this parameter for a fixed-configuration line, the value should be unique for each terminal on the line. The network software performs an additional match while identifying the terminal in the configuration file. Any terminal accessing the line must use a cluster address you specify or it cannot use the network.

The CA parameter can have the following value:

AUTOREC

Indicates that your site-written TIP should determine any needed cluster address during automatic recognition of the terminal (valid only if the AUTO or XAUTO parameter is used in the corresponding LINE or GROUP statement); using this value is equivalent to omitting the parameter.

If you specify the value of AUTOREC, your site-written TIP accepts any cluster address it recognizes. If you omit the CA parameter, the default is unknown and possibly undefined.

CO PARAMETER

The CO parameter specifies the configuration ordinal for all the devices accessing the line through one terminal. If you supply this parameter for an automatic recognition line, the value should be unique for each terminal that can use the line. The network software performs an additional match while identifying the terminal in the configuration file. Any terminal accessing the line must use a configuration ordinal you specify or it cannot use the network.

The CO parameter can have the following values:

AUTOREC

> Indicates that your site-written TIP should determine the configuration ordinal during automatic recognition of the terminal (valid only if the AUTO or XAUTO parameter is used in the corresponding LINE or GROUP statement); using this value is equivalent to omitting the parameter.

$1 \leq conord \leq 255$

> Indicates the decimal configuration ordinal of the terminal you are defining.

If you specify the value of AUTOREC, your site-written TIP accepts any configuration ordinal it recognizes. If you omit the CO parameter, the default value is unknown and possibly undefined.

COLLECT PARAMETER

The COLLECT parameter specifies whether your site-written TIP accepts charges for incoming calls from X.25 terminals connected through a packet assembly/disassembly service. This stand-alone parameter is optional.

The COLLECT parameter can have the following values:

NO   Your site rejects any incoming call when the X.25 network indicates that charges will occur.

YES   Your site accepts charges from the X.25 network.

If you specify the COLLECT parameter without a value, the value of YES is used. If you omit the COLLECT parameter, the default value is unknown and possibly undefined.

CSET PARAMETER

The CSET parameter specifies the code and character set of the terminal. This parameter is optional. When you declare a value for CSET, it must be one of the following reserved words:

APLBP   ASCII code set with bit-pairing APL character set.

APLTP   ASCII code set with typewriter-pairing character set.

ASCII   ASCII code and character set, or ASCII code and a different character set (some terminals permit other character sets with this code set).

AUTOREC The default code set for the terminal class or that is determined during automatic recognition of the terminal device; using this value is equivalent to omitting the parameter.

BCD   CDC External BCD code set with external BCD character set or 63-character ASCII subset (some terminals permit either character set with this code set).

CORAPL IBM Correspondence code set with APL character set.

CORRES IBM Correspondence code and character set.

CSET15 Site-defined code and character set, identified within the network software as character set number 15.

EBCD   IBM Extended BCD code and character set.

EBCDAPL IBM Extended BCD code set with APL character set.

EBCDIC IBM Extended Binary Coded Decimal Interchange Code and character set.

If you specify CSET for a device on an automatic recognition line, the network software can perform an additional match while identifying the terminal in the configuration file. If you omit CSET, no additional match is performed for automatic recognition lines and the default is unknown and possibly undefined for fixed-configuration lines.

EOF PARAMETER

The EOF parameter indicates whether and end-of-file indicator is sent to line printer devices to mark the position in the data where an end-of-file was read from disk. The end-of-file indicator is an ESC V sequence.

Possible values for this parameter are:

NO   Indicates that an end-of-file indicator is not sent to line printer devices.

YES   Indicates that an end-of-file indicator is sent to line printer devices.

If you omit this parameter, the default is unknown and possibly undefined.

MREC PARAMETER

The MREC parameter indicates the maximum number of records that each device within the terminal can receive in each transmission block. This number determines how the network software blocks data into a transmission block for output.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP

Indicates that your site-written TIP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$1 \leq mrec \leq 7$

Indicates the number of records required by the terminal in each block exchanged with the network.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

NCIR PARAMETER

The NCIR parameter specifies the decimal number of X.25 virtual circuits of the same subTIPtype you are defining. This parameter should be used only for switched virtual circuit terminal definitions.

If you specify this parameter, the value you use should be less than or equal to the NSVC value declared or used on the corresponding LINE statement. The NCIR parameter has the following range of values:

$1 \leq numcir \leq 255$

If you omit this parameter, the default is unknown and possibly undefined.

NEN PARAMETER

The NEN parameter specifies the decimal number of X.25 virtual circuits of the same subTIPtype that are initially enabled. This parameter is optional; it is usually used only for switched virtual circuit terminal definitions.

If you specify this parameter, the value you use should be less than or equal to the NCIR value declared or used on the same TERMINAL or TERMDEV statement. The NEN parameter has the following range of values:

$1 \leq encir \leq 255$

If you omit this parameter, the default value is unknown and possibly undefined.

PAD PARAMETER

The PAD parameter specifies a string of 4 to 64 hexadecimal digits representing the ASCII equivalent of PAD parameter fields. Substrings should consist of groups of 4 hexadecimal digits (2 ASCII characters).

The PAD parameter is optional. If you omit the PAD parameter, the default is unknown and possibly undefined.

RIC PARAMETER

The RIC parameter indicates whether the terminal has restricted interactive capabilities. This parameter is optional.

A terminal can have restricted interactive capabilities for many reasons; for example, perhaps no console device actually exists and interactive dialog must occur through a site-defined device.

The effect this parameter has depends on the application program the terminal uses. For example, RBF does not output the READY prompt and does not require a GO command to enable any defined passive devices when RIC is specified.

Possible values for this parameter are:

NO   Indicates that the terminal has full interactive capabilities.

YES   Indicates that the terminal has limited interactive capabilities.

If you omit the RIC parameter, the default is unknown and possibly undefined.

TSPEED PARAMETER

The TSPEED parameter specifies the baud rate of the terminal. This parameter is optional.

If you specify a TSPEED value, the network software can perform an additional match while identifying the terminal in the configuration file. Any terminal accessing the communication line using the speed you specify might match the terminal definition.

If AUTO is specified, the TSPEED parameter can have the following values:

110   134   150   300   600   1200   2400
AUTOREC

If XAUTO is specified, the TSPEED parameter can have the following values:

600   1200   2400   4800   9600   AUTOREC

If you omit the TSPEED parameter or use the value AUTOREC, your site-written TIP determines the baud rate when it performs automatic recognition for the terminal (a CDC-defined TIP accepts any rate valid for the communication line).

W PARAMETER

The W parameter specifies the X.25 packet level window size for the virtual circuit you are defining. This parameter is optional.

The packet level window is the maximum number of unacknowledged packets your site-written TIP can send before it suspends further output to the terminal using the virtual circuit. This parameter has the following range of values:

$1 \leq pacwndw \leq 7$

If you omit the W parameter, the default is unknown and possibly undefined.

DEVICE DEFINITIONS

You must provide one TERMDEV or DEVICE statement for each device that can access the communication line through the terminal. If you use a TERMDEV statement, you cannot use a DEVICE statement for the same terminal.

Each TERMINAL statement can have 255 DEVICE statements. Figures 7-5 and 7-6 present the formats of the TERMDEV and DEVICE statements for terminals on site-defined protocol lines.

The following TERMDEV and DEVICE parameters provide the device definition for a terminal:

AB	DT	PA
ABL	EBO	PG
AUTOCON	EBR	PL
BR	EBX	PRI
BS	ELO	PW
B1	ELR	P90 through P99
B2	ELX	RTS
CI	EP	SDT
CN	HD	STREAM
CP	HN	TA
CT	IC	UBL
DBL	IN	UBZ
DBZ	LI	XBZ
DI	LK	XLC
DLC	MCI	XLTO
DLTO	MLI	XLX
DLX	OC	XLY
DO	OP	

The DT parameter is described first because it is described first in sections 4 through 6. The other parameters are described in alphabetical order.

DT PARAMETER

The DT parameter specifies the device type being defined. You can declare either a CDC-defined device type or a site-defined device type.

This parameter is optional. When you specify the DT parameter, the following values are valid:

AP	Identifies an application-to-application connection.
CON	Identifies a CDC-defined console device.
CP	Identifies a CDC-defined card punch.
CR	Identifies a CDC-defined card reader.
DT12	Identifies a site-defined device with no predefined characteristics, using the device type number 12 within NAM.
LP	Identifies a CDC-defined line printer.
PL	Identifies a CDC-defined plotter.

If you omit the DT parameter, the default is unknown and possibly undefined.

AB PARAMETER

The AB parameter specifies the character to be used to abort an output block. When the terminal user enters this character as the only character on a line, your site-written TIP should discard the block of data being transmitted to the terminal.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

```
device:  DEVICE[,DT=devtyp,SDT=subdt,TA=trmaddr,ABL=abl,DBZ=dwnlsiz,UBZ=upbsize,]

    [DBL=dwnblim,UBL=upblim,XBZ=xmitsiz,DO=devord,STREAM=streamno,AUTOCON[=yn1],]

    [PRI[=yn2],DI[=yn3],HN=node,HD[=yn4],LK=yn5,AB=ab,BR=br,CP=cp,BS=bs,B1=b1,B2=b2,CI=ci,]

    [CN=cn,CT=ct,DLC=dlc,DLTO=dlto,DLX=dlx,EBX=ebx,EBR=ebr,EBO=ebo,ELX=elx,ELR=elr,]

    [ELO=elo,EP=ep,XLC=xlc,XLTO=xlto,XLX=xlx,XLY=xly,IC=ic,IN=in,LI=li,OP=op,]

    [OC=oc,PA=pa,PG=pg,PL=pl,PW=pw,RTS[=yn6],MCI=mc,MLI=ml,P90=fv90,...,P99=fv99].


device      The element name of the terminal device being defined.  This name can be one through seven
            characters long.  The first character must be a letter; the other characters can be
            letters or digits.  If this statement is used within the set following a GROUP statement
            or as a switched virtual circuit definition, the device name cannot be longer than five
            characters.  This is the name used by the host or NPU operator to monitor and control the
            device.  This name is required; there is no default value.

All other parameters are described in the text.
```

Figure 7-6. DEVICE Statement Format for Site-Defined Communication Line Protocols

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the AB parameter or specify the value CCP, the default is unknown and possibly undefined.

ABL PARAMETER

The application block limit (ABL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. The value you choose should keep the terminal busy for 2 seconds by maintaining that number of outstanding blocks of the size specified by the DBZ parameter.

This parameter is optional. The ABL parameter has the following range of values:

$$1 \leq abl \leq 7$$

The value you declare should be greater than or equal to the downline block limit (DBL parameter value) of the terminal. The host queues abl - dbl blocks; the NPU queues dbl blocks. An ABL value significantly larger than the DBL value causes NAM to use more host memory.

If you omit the ABL parameter, the default value is unknown and possibly undefined.

AUTOCON PARAMETER

The AUTOCON parameter is a stand-alone keyword that determines whether the TIP should automatically connect the console device and any associated batch devices to the selected host node. This parameter is valid only for console devices.

If all logical links terminating in the NPU being defined are to the same host (all HNAME values are equal), and the NDL processor provides a default host node, then the NDL processor also sets AUTOCON unless AUTOCON=NO is specified.

This parameter and its values are optional. You can specify either of the following values:

NO   Indicates that the TIP should not attempt automatic connection.

YES   Indicates that the TIP should attempt automatic connection.

If you specify this parameter without a value, the value of YES is used. If you omit this parameter, the default is unknown and possibly undefined.

BR PARAMETER

The BR parameter determines whether your site-written TIP associates data control functions with the pressing of the break function key. The break function key is usually labeled BREAK, ATTN, INTER, INTERRUPT, or something similar.

If the break function key is not associated with data control, pressing the break function key only interrupts output. After input is completed, output probably resumes at the next character.

If the break function key is associated with data control, the following probably occurs when the terminal user presses the break function key:

If output is in progress, the TIP discards all output queued for the device.

If output is in progress or the terminal is idle, the TIP sends a user break 1 message upline (the function also associated with the character defined by the B1 parameter).

After input occurs, output resumes with the next block transmitted from the host.

The BR parameter is optional. If you specify this parameter, the following values are recognized:

CCP   Indicates that the TIP should perform the default action appropriate for the terminal class. Using this value is equivalent to omitting the parameter.

NO   Indicates that the break function key has no control functions.

YES   Indicates that the break function key has control functions.

If you omit the BR parameter or specify the value CCP, the default is unknown and possibly undefined.

BS PARAMETER

The BS parameter specifies the character to be used for the single-character deletion function. When the terminal user enters this character, your site-written TIP discards the preceding character transmitted by the device (unless the character has already been sent to the host). The character code used for the deletion function does not produce a physical backspace at the device unless the device also recognizes the code for that function.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the BS parameter or specify the value CCP, the default is unknown and possibly undefined.

B1 PARAMETER

The B1 parameter specifies the character to be used as a user break 1 indicator. When the terminal user enters this character as the only character on a line, your site-written TIP discards the block of data being transmitted to the terminal and sends a user break 1 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 1 message as a job step interrupt from the terminal user.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

  00, 01, or 02 (characters NUL, SOH, or STX)

  3D, 7F, or 20 (characters =, DEL, or space)

  30 through 39 (characters 0 through 9)

  41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B1 parameter or specify the value CCP, the default is unknown and possibly undefined.

B2 PARAMETER

The B2 parameter specifies the character to be used as a user break 2 indicator. When the terminal user enters this character as the only character on a line, your site-written TIP discards the block of data being transmitted to the terminal and sends a user break 2 message upline to the connected application program.

The effect this parameter has depends on the application program. For example, IAF interprets the user break 2 message as a job step termination from the terminal user.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

  00, 01, or 02 (characters NUL, SOH, or STX)

  3D, 7F, or 20 (characters =, DEL, or space)

  30 through 39 (characters 0 through 9)

  41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the B2 parameter or specify the value CCP, the default value is unknown and possibly undefined.

CI PARAMETER

The CI parameter specifies the number of idle characters to insert in the downline data after a carriage return. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically return the carriage of the device to its left margin for the next line of output.

If you declare this parameter, you must use the reserved word CCP or a value in the following range:

$$0 \le ci \le 127$$

If you omit the CI parameter or specify the value CCP, the default value is unknown and possibly undefined.

CN PARAMETER

The CN parameter specifies the character to be used to abort (cancel) an input message. When the device user enters this character as the last character on a line, your site-written TIP and the network software discard the current logical line when it is transmitted from the terminal.

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

  00, 01, or 02 (characters NUL, SOH, or STX)

  3D, 7F, or 20 (characters =, DEL, or space)

  30 through 39 (characters 0 through 9)

  41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CN parameter or specify the value CCP, the default value is unknown and possibly undefined.

CP PARAMETER

The CP parameter specifies whether your site-written TIP should send a cursor positioning response when the terminal user enters an end-of-line or end-of-block character, such as a CR or

EOT, or the end-of-packet indicator. The end-of-line and end-of-block characters are the current values of the ELX and ELB parameters.

The cursor positioning response used is probably determined by the the current setting of the ELR and EBR parameters. The usual response moves the cursor to the beginning (left margin) of the next line on the screen; this is equivalent to a carriage return and linefeed operation.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that your site-written TIP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that no cursor positioning response should occur.

YES   Indicates that the cursor should be repositioned.

If you omit this parameter or specify the value of CCP, the default value is unknown and possibly undefined.

CT PARAMETER

The CT parameter specifies the character to be used as a flag for a terminal definition command. When the terminal user enters this character as the first character on a line, your site-written TIP should interpret the line as a command. Among other functions performed by CDC-written TIPs, commands are provided to determine or change the values you have established for the following parameters:

AB	CI	DLTO	EP	OC	PW	XLY
BR	CN	DLX	HN	OP	TC	
BS	CP	EBR	IC	PA	XLC	
B1	CT	EBX	IN	PG	XLTO	
B2	DLC	ELX	LI	PL	XLX	

If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit the CT parameter or specify the value CCP, the default value is unknown and possibly undefined.

DBL PARAMETER

The downline block limit (DBL) parameter specifies the number of downline blocks that can be outstanding (unacknowledged) between the host computer and this terminal. The value you choose determines how many blocks of data the NPU queues from the total number of outstanding blocks (ABL parameter value) of the size specified by the DBZ parameter.

This parameter is optional. The DBL parameter has the following value range:

$$1 \leq dbl \leq 7$$

The value you declare should be less than or equal to the application block limit (ABL parameter value) of the terminal. The host queues abl - dbl blocks; the NPU queues dbl blocks. Small DBL values use less NPU memory but cause slower data transfers.

If you omit the DBL parameter, the default value is unknown and possibly undefined.

DBZ PARAMETER

The downline block size (DBZ) parameter specifies the maximum number of character bytes each downline block can contain. The value you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host. Larger DBZ values cause fewer disk accesses by the host. This value can be used by the application programs to divide downline messages into blocks.

This parameter is optional. The DBZ parameter has the following value range:

$$1 \leq dbz \leq 2043$$

The value you declare should be chosen together with the value used for the DBL parameter.

If you omit the DBZ parameter, the default value is unknown and possibly undefined.

DI PARAMETER

The DI parameter is a stand-alone keyword that specifies whether a device is initially enabled or disabled. An enabled device is configured and serviced as soon as the communication line becomes active. A disabled device is neither configured nor serviced when the line becomes active.

If you initially disable the device, a host or NPU operator can change the device status to enabled. This change can be made only when the line becomes active (when a call is received on a dialup line, or when communications are established on a hardwired line).

This parameter is optional. If you specify the DI parameter, you must use one of the following values:

NO   Indicates that the device is initially enabled.

YES   Indicates that the device is initially disabled.

If you specify the DI parameter without a value, the default value of YES is used. If you omit this parameter, the default is unknown and possibly undefined.

DLC PARAMETER

The DLC parameter indicates the maximum number of characters that can be input in each single-message transparent mode upline message from this device. After the terminal user or the application program changes the device's input mode to transparent from normalized, your site-written TIP forwards a message block to the application and changes back from transparent mode to normalized mode when the device transmits the indicated number of characters.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP

Indicates that the TIP should use the default number of characters appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$1 \leq dlc \leq 4095$

Indicates the maximum decimal number of characters that the terminal can transmit as a single transparent mode message.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

DLTO PARAMETER

The DLTO parameter specifies whether a 200- to 400-millisecond timeout on the communication line ends single message transparent mode input. After the terminal user or the application program changes the device's input mode to transparent from normalized, your site-written TIP forwards a message block to the application and changes back from transparent mode to normalized mode when the device stops transmitting for the indicated timeout period.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that a timeout does not end transparent mode input.

YES   Indicates that transparent mode input ends when a timeout occurs.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

DLX PARAMETER

The DLX parameter indicates the hexadecimal value of the character code that should end input of a single-message transparent mode message from the device. After the terminal user or the application program changes the device's input mode to transparent from normalized, your site-written TIP forwards a message block to the application and changes back from transparent mode to normalized mode when the device transmits this character.

You should select transparent mode input delimiters with care. The character code you declare as the DLX value must be the code of a character that the physical device can input. If the delimiter is a character that cannot be input (either because of device hardware limitations or because of the PA parameter value declared for the device), then the Terminal Interface Program cannot terminate transparent mode input and the device will be trapped in that mode of operation once it has begun.

This parameter is optional. The value that is valid for a given character depends on the character code set used by the terminal. You can use the following values:

CCP

Indicates that the TIP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \leq dlx \leq FF$

Indicates the hexadecimal code that ends single message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F might not be seen by the TIP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

DO PARAMETER

The DO parameter specifies the device ordinal of the device within the terminal. This device ordinal provides a unique identifier to an application program for the device when more than one device of the same type is part of the terminal.

This parameter is optional. However, the Remote Batch Facility requires a device ordinal for each passive device. Passive devices with a terminal class of 200UT or 734 implicitly have device ordinals of 1. For passive terminals with a TC value of 714 or 714X, the device ordinal must be unique if the device communicates with RBF. (An implicit default value of 1 exists when there is only one device of each type owned by the same owning console.) The DO parameter is the only device definition parameter with this uniqueness requirement.

If you specify this parameter, you must use a value in the range:

$$1 \leq \text{devord} \leq 7$$

If you omit this parameter, the default is unknown and possibly undefined.

EBO PARAMETER

The EBO parameter indicates the event that identifies an end-of-block condition. During input, messages are transmitted upline as soon as an end-of-line code (or message-forwarding code sequence) or an end-of-block condition is detected.

When some devices perform input, they use code sequences or events to indicate the end-of-line and end-of-block conditions. (For example, the end-of-block sequence from a mode 4 device is ESC A.) Code sequences and events cannot be declared as values for the EBX parameter. The EBO parameter allows you to set the end-of-block condition to one of these code sequences or events.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that your site-written TIP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

EL   Indicates that the default end-of-line character code or code sequence for the terminal class should be the end-of-block condition indicator. This value causes the network to treat a single input message the same as a multiple-message block (output to the terminal can occur after each input message, instead of after all messages in a block).

If you omit this parameter or specify the value CCP, the end-of-block condition indicator is probably the value declared for the EBX parameter. If no value is declared for the EBX parameter, the end-of-block condition might be undefined.

EBR PARAMETER

The EBR parameter indicates the appropriate cursor positioning response whenever a block of input ends. A block of input ends when your site-written TIP detects the currently defined end-of-block condition, which is either a character code or the end-of-packet sequence. The end-of-block condition is determined by the EBO or EBX parameter value. Whether cursor positioning occurs probably is determined by the CP parameter value.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should perform the default response appropriate for the terminal class; using this value is equivalent to omitting the parameter.

CL   Indicates that the TIP should perform a carriage return and linefeed operation (move the cursor left and down to the beginning of the next physical line).

CR   Indicates that the TIP should perform a carriage return operation (move the cursor left to the beginning of the current physical line).

LF   Indicates that the TIP should perform a linefeed operation (move the cursor down to the next physical line).

NO   Indicates that the TIP should not reposition the cursor.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

EBX PARAMETER

The EBX parameter identifies the character code that ends a block of messages (logical lines) from the device. When your site-written TIP detects this code in input, it forwards the last message of the block upline.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

ELO PARAMETER

The ELO parameter indicates the event that identifies an end-of-line condition (an end-of-logical line or message-forwarding signal). During input, a message is transmitted upline as soon as an end-of-line code (or message-forwarding code sequence) or an end-of-block condition is detected.

When some devices perform input, they use code sequences or events to indicate the end-of-line and end-of-block conditions. Code sequences and events cannot be declared as values for the ELX parameter. The ELO parameter allows you to set the end-of-line indicator to one of these sequences or events.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

EB   Indicates that the current end-of-block character code or code sequence for the terminal class should be the end-of-line condition indicator. This value causes any default end-of-line codes or code sequences to be sent upline as data within a single message.

If you omit this parameter or specify the value CCP, the end-of-line condition indicator probably is the value declared for the ELX parameter. If no value is declared for the ELX parameter, the end-of-line condition indicator is unknown and might be undefined.

ELR PARAMETER

The ELR parameter indicates the appropriate cursor positioning response whenever a message (logical line) of input ends. A logical line of input ends when your site-written TIP receives the currently defined end-of-line character (ELX parameter value). Whether cursor positioning occurs probably is determined by the CP parameter value.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should perform the default response appropriate for the terminal class; using this value is equivalent to omitting the parameter.

CL   Indicates that the TIP should perform a carriage return and linefeed operation (move the cursor left and down to the beginning of the next physical line).

CR   Indicates that the TIP should perform a carriage return operation (move the cursor left to the beginning of the current physical line).

LF   Indicates that the TIP should perform a linefeed operation (move the cursor down to the next physical line).

NO   Indicates that the TIP should not reposition the cursor.

If you omit this parameter or specify the value CCP, the default is unknown and might be undefined.

ELX PARAMETER

The ELX parameter identifies the character code that ends a message (logical line) from the device. When your site-written TIP detects this code in input, it should forward the message upline.

This parameter is optional. If you specify a value, it must be either the reserved word CCP or a hexadecimal value for the 7-bit ASCII code that

represents the character to be used (table A-2 in appendix A lists all of the codes in the 7-bit ASCII code set).

You cannot use any of the following values:

00, 01, or 02 (characters NUL, SOH, or STX)

3D, 7F, or 20 (characters =, DEL, or space)

30 through 39 (characters 0 through 9)

41 through 5A or 61 through 7A (alphabetic characters A through Z or a through z)

If the terminal does not use an ASCII character set, you should choose a character that has a unique translation to a character in the 7-bit ASCII code set.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

EP PARAMETER

The echoplex (EP) parameter causes or prevents the echoing of input characters back to the output mechanism of the device. Echoplexing is normally required when the device operates in full-duplex mode and no echoing is performed by hardware between the device input hardware and your site-written TIP.

This parameter is optional. If you specify the EP parameter, you must use one of the following values:

CCP   Indicates that the TIP should echo input if that is the appropriate default action for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that the TIP should not echo input characters.

YES   Indicates that the TIP should echo input characters.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

HD PARAMETER

The HD parameter specifies whether or not the full host availability display (HAD) is presented to the terminal user. This parameter is optional and is valid only for console devices.

If you specify this parameter, you must use one of the following values:

NO   Indicates that the full host availability display is not presented to the terminal user. The terminal user receives only the host status message and the prompt message.

YES   Indicates that the full host availability display is presented to the terminal user.

If you omit this parameter, the default is unknown and possibly undefined.

HN PARAMETER

The HN parameter identifies the node number of the host that a console device and any associated batch devices are connected to unless another path is selected by the terminal user.

This parameter is optional and is valid only for console devices. If you declare this parameter, you must use one of the following values:

NONE

Indicates that no node number is specified. This has the same effect as not declaring the parameter. A value of NONE cannot be declared if AUTOCON is also specified and there are logical links to more than one host terminating in the NPU being defined.

$1 \leq hn \leq 255$

Indicates the node number of the host that the console device and any associated batch devices are to be connected to.

The value declared for the HN parameter must be the same as the NODE value in a COUPLER statement within the same network definition. If all the COUPLER statements for all the logical links to the NPU being defined have equal HNAME parameters, the NDL processor provides a default host node. The default host node is the last coupler specified. If the HN parameter is not specified, the terminal user must select a host before a connection can be made. If the AUTOCON parameter is specified, then the HN parameter must also be specified.

IC PARAMETER

The IC parameter specifies whether or not the input mechanism of the device supports an ASCII DC3 code (X-OFF character) as a signal to stop input and an ASCII DC1 code (X-ON character) as a signal to resume input. Your site-written TIP transmits these codes to control input flow when the device can support them.

This parameter allows you to configure a device so that input from an intelligent terminal such as a personal computer or from a cassette mechanism can be interrupted and restarted as needed without terminal user intervention. The TIP sometimes needs to suspend input because the volume of network traffic has temporarily used all available storage space.

You should be careful that this parameter is appropriately defined for the device hardware that actually uses the line. These codes are used for many purposes by device manufacturers; receiving either code can have effects other than resuming or stopping input.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should use the default appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that the TIP cannot use X-ON and X-OFF characters to control input.

YES   Indicates that the TIP can use X-ON and X-OFF characters to control input.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

IN PARAMETER

The IN parameter identifies the input mechanism and the input message blocking of the device. (Terminal users might have a similar command that also defines whether input is in transparent mode or normalized mode; you cannot configure a device to begin accessing the network in transparent mode.)

As the input device, you can specify either:

The keyboard

The paper tape reader

As the input transmission mode, you can specify either:

Normalized line mode (one message or logical line per block)

Normalized block mode (one or more logical lines collected into a block before it is transmitted. Cursor positioning at linefeed and end-of-line is not performed and output is not sent until end-of-block is reached.)

This parameter is optional. Possible values are:

BK   Indicates keyboard input in block mode.

CCP   Indicates that the TIP should use the default options appropriate for the terminal class; using this value is equivalent to omitting the parameter.

KB   Indicates keyboard input in line mode.

PT   Indicates paper tape input in block mode (an X-OFF character functions as a fixed end-of-block condition indicator).

When the IN value is set to PT, your TIP should send an ASCII DC1 code (X-ON character) to start the paper tape reader, if the device supports that interpretation of the code. The X-ON code is issued after the end of a message is output to the device and the message empties the TIP's output queue for the device.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

LI PARAMETER

The LI parameter specifies the number of idle characters to insert in the downline data after a linefeed code. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically move the carriage of the device to its next line for output.

This parameter is optional. If you declare this parameter, you must use the reserved word CCP or a value in the following range:

$$0 \leq li \leq 127$$

If you omit the LI parameter or specify the value CCP, the default is unknown and possibly undefined.

LK PARAMETER

The LK parameter specifies whether unsolicited messages from the NPU or host operator can appear at the terminal. This parameter is optional.

If you specify this parameter, you must use one of the following values:

CCP   Indicates that your site-written TIP should use the default mode appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that unsolicited messages should be delivered to the terminal as soon as received by your site-written TIP.

YES   Indicates that unsolicited messages should be discarded (locked out).

If you omit this parameter or specify the value CCP, the default value is unknown and possibly undefined.

MCI PARAMETER

The MCI parameter specifies the delay after a carriage return is output. This delay is in 4-millisecond increments. This value is used by CCP to compute the number of idle characters output.

This parameter is optional. The MCI parameter has the following value range:

$$0 \leq mci \leq 250$$

If you omit the MCI parameter, the default value is unknown and possibly undefined.

MLI PARAMETER

The MLI parameter specifies the delay after a line feed is output. This delay is in 4-millisecond increments. This value is used by CCP to compute the number of idle characters output.

This parameter is optional. The MLI parameter has the following value range:

$$0 \leq mli \leq 250$$

If you omit the MLI parameter, the default value is unknown and possibly undefined.

OC PARAMETER

The OC parameter specifies whether or not the output mechanism of the device sends an ASCII DC3 code (X-OFF character) as a signal for your site-written TIP to interrupt output and an ASCII DC1 code (X-ON character) as a signal for the TIP to resume output. The TIP might accept these codes for control of output flow by devices that must periodically interrupt output to perform such functions as emptying buffers to offline storage devices. DC1 and DC3 codes used in this manner are probably discarded by the TIP.

You should only use this parameter when the device actually using the line requires it. The DC1 and DC3 codes can be sent for many purposes by terminal users; either code in input might be intended as data for the host application program.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should use the default appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that the TIP should ignore X-ON and X-OFF characters in input; output control by the device is unnecessary.

YES   Indicates that the TIP should recognize X-ON and X-OFF characters in input as output control by the device.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

OP PARAMETER

The OP parameter identifies the output mechanism, page width processing requirement, and page length processing requirement of the device. Your site-written TIP might support these output mechanisms:

A console screen

A hardcopy mechanism, such as a printer

A paper tape punch

The TIP might perform one of these processing actions when a finite page width is reached:

Inserts the codes appropriate to return the cursor or carriage to the beginning of the next physical line.

Inserts no codes, which performs no action.

If the device has a page width of 0, the second choice is probably used. Refer to the PW Parameter description.

Your site-written TIP might perform one of these processing actions when a finite page length is reached:

Inserts the codes appropriate to cause the cursor or carriage to move to the next page (clear screen, formfeed, and so forth).

Inserts no codes, which performs no action.

If the device has a page length of 0, the second choice is probably used. Refer to the PL Parameter description.

This parameter is optional. If you specify the OP parameter, you must use one of the following values:

CCP   Indicates that the TIP should use the default options appropriate for the terminal class; using this value is equivalent to omitting the parameter.

DI   Indicates that the console display screen is used, no codes should be inserted when page width is reached (the device does its own line folding), and codes should be inserted when page length is reached.

PR   Indicates that a hardcopy printer is used, codes should be inserted when page width is reached (the device does not do its own line folding), and codes should be inserted when page length is reached.

PT   Indicates that a paper tape punch is used, lines should be folded, but codes should not be inserted when page length is reached, and page wait is not effective.

When the OP value is set to PT, your TIP might send an ASCII DC1 code (X-OFF character) at the end of the message with postprint format control.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

PA PARAMETER

The PA parameter indicates the processing your site-written TIP should perform for the parity bit within each character byte of upline or downline data. The TIP might process the parity bit of all upline character codes in one of five ways:

Ignores the input byte parity bit and sets it to zero after input. Converts the byte to an ASCII code if the device is operating in normalized mode, then forwards the byte upline to the application program. This is called zero parity processing.

Checks the input byte parity bit to ensure that it is set to zero or one, as necessary to give the byte an even number of bits; odd settings are indicated to the receiving application program as parity errors. Sets the parity bit to zero after input, converts the byte to an ASCII code if the device is operating in normalized mode, then forwards the byte upline to the application program. This is called even parity processing.

Checks the input byte parity bit to ensure that it is set to zero or one, as necessary to give the byte an odd number of bits; even settings are indicated to the receiving application program as parity errors. Sets the parity bit to zero after input, converts the byte to an ASCII code if the device is operating in normalized mode, then forwards the byte upline to the application program. This is called odd parity processing.

Ignores the input byte parity bit. If the device is operating in normalized mode, sets the parity bit to zero, converts the byte to an ASCII code, and then forwards the byte upline to the application program. If the device is operating in transparent mode, forwards the 8 bits unchanged to the application (for devices that send 8-bit bytes, this allows the application program to receive bit 7 as data). This is called no parity processing.

Ignores the input byte parity bit. If the device is operating in normalized mode, sets the parity bit to zero, converts the byte to an ASCII code, and then forwards the byte upline to the application program. If the device is operating in transparent mode, ignore the parity bit when checking for transparent mode input delimiters. Forward the 8 bits unchanged to the application in the same way as for no parity. This is called ignore parity processing.

The TIP might also process the upper bit (bit 7) of all downline character code bytes in one of four ways:

If the device is operating in normalized mode, the TIP converts the lower seven bits of the byte, if necessary, to the code set used by the device. The parity bit in the output byte is set to zero, regardless of the setting of bit 7 in the downline byte. This is called zero parity processing.

If the device is operating in normalized mode, the TIP converts the lower seven bits of the byte, if necessary, to the code set used by the device. The parity bit in the output byte is set to zero or one, as necessary to give the byte an even number of set bits; the setting of bit 7 in the downline byte is ignored. This is called even parity processing.

If the device is operating in normalized mode, the TIP converts the lower seven bits of the byte, if necessary, to the code set used by the device. The parity bit in the output byte is set to zero or one, as necessary to give the byte an odd number of set bits; the setting of bit 7 in the downline byte is ignored. This is called odd parity processing.

If the device is operating in normalized mode, the TIP converts the lower seven bits of the byte, if necessary, to the code set used by the device and the parity bit in the output byte is set to zero, regardless of the setting of bit 7 in the downline byte. If the device is operating in transparent mode, the parity bit in the output byte is unchanged from the setting of bit 7 in the downline byte (for devices that can receive 8-bit bytes, this allows the application program to use bit 7 as data). This is the method CCP uses for both no parity and ignore parity processing.

You can specify one of these input and output processing options to match the parity bit input processing performed by the device and the output processing expected by it. If the device will operate in transparent mode, you must specify either the no parity or the ignore parity option for an application program and the device to exchange hexadecimal codes with values between 7F and FF (using bit 7 as data).

No parity and ignore parity processing differ only in the way that a transparent input delimiter is recognized. For both PA=N and PA=I, all 8 bits are treated as data during transparent input and output. For PA=N, a transparent input delimiter is recognized when the 8-bit byte input from the device matches the 8-bit byte specified as the delimiter. For PA=I, a transparent input delimiter is recognized when the bottom 7 bits of the byte input from the device match the bottom 7 bits of the specified delimiter; the parity bit is ignored.

This parameter is optional. If you specify the PA parameter, you must use one of the following values:

CCP   Indicates that the TIP performs the parity processing appropriate as a default for the terminal class; using this value is equivalent to omitting the parameter.

E   Indicates that the TIP should perform even parity processing for both input and output bytes.

I   Indicates that the TIP performs ignore parity processing for both input and output bytes.

N   Indicates that the TIP should perform no parity processing for both input and output bytes.

O   Indicates that the TIP should perform odd parity processing for both input and output bytes.

Z   Indicates that the TIP should perform zero parity processing for both input and output bytes.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

PG PARAMETER

The PG parameter specifies whether your site-written TIP should wait at each output page boundary for terminal user acknowledgment before it displays the next page of data. In certain situations, page waiting can occur other than at page boundaries; the TIP might produce a prompting message (OVER..) when this type of page waiting occurs. The user's response to page waiting is entry of a line, usually empty.

The TIP probably views a new page as beginning at the start of each downline message. The TIP calculates the length of a page from the current values of the page width and page length (see PW and PL Parameter descriptions). If the page width is infinite (PW=0), a page consists of one line less than the number of logical lines specified as the page length. If the page width is finite (PW is nonzero), a page consists of one line less than the number of physical lines specified by the page length; the TIP calculates the number of physical lines by dividing each logical line into units less than or equal to the page width.

This parameter is optional. If you specify this parameter, you must use one of the following values:

CCP   Indicates that the TIP should use the default setting appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that page waiting should not occur.

YES   Indicates that page waiting should occur.

When you specify PG=YES, you should also specify a nonzero value for the PL parameter. A TIP cannot perform page waiting at the boundaries of infinitely long pages (PL=0).

If you omit the PG parameter or specify a value of CCP, the default is unknown and possibly undefined.

PL PARAMETER

The PL parameter specifies the number of physical lines per page of output for the device. If the device is configured for page waiting (PG parameter), any message containing more lines of output than the page length should be interrupted by your site-written TIP for a page waiting response from the terminal user. The interruption probably occurs after line pl − 1 is output. If the device uses a hardcopy output mechanism (OP of PR, such as is normal for terminal classes 2741 or M33), your site-written TIP should insert formfeed codes at page length boundaries.

This parameter is optional. If you specify the PL parameter, you must use one of the following values:

CCP

     Indicates that the TIP should use the default page length appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

     Indicates an infinite page length. An infinite page length means that no page waiting occurs and no form-feed codes are inserted.

$8 \leq pl \leq 255$

     Indicates the number of physical lines per page.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

PRI PARAMETER

The PRI parameter is a stand-alone keyword that indicates whether data to or from the device is to have traffic priority over that to or from other devices in the network. Data to or from a device that does not have traffic priority is suspended to relieve network congestion before data to or from terminals with traffic priority is suspended. Interactive terminals should usually be given traffic priority over batch devices.

This parameter is optional. If you specify the PRI parameter, you must use one of the following values:

NO   Indicates that the device should not have data traffic priority.

YES   Indicates that the device should have data traffic priority.

If you declare this parameter without a value, the value of YES is used. If you omit this parameter, the default is unknown and possibly undefined.

PW PARAMETER

The PW parameter defines the number of characters per physical line of output for this device. This physical line length is also called the page width.

For console devices (DT of CON) with a printer defined as the output mechanism (OP of PR), output lines longer than pw characters are divided into lines of pw or fewer characters each. For console devices with a display defined as the output mechanism (OP of DI), output lines longer than pw characters are not divided but are counted as more than one line for page length calculations; if the PW value for a device is inappropriate, loss of visual fidelity can occur (the application program might divide the data into lines that are too short or too long for the screen's capacity, instead of allowing the terminal to wrap lines when needed).

This parameter is optional. If you specify the PW parameter, you must use one of the following values:

CCP

Indicates that CCP should use the default value appropriate for the terminal class; using this value is equivalent to omitting the parameter.

0

Indicates that the device has an infinite page width. An infinite page width probably means that physical line length has no effect on output formatting.

$20 \leq pw \leq 255$

Indicates that the device can support physical lines no longer than the indicated decimal value.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

P90 THROUGH P99 PARAMETERS

These ten parameters indicate the hexadecimal field value to use for the corresponding field number within configuration information transmitted for this device to your site-written Terminal Interface Program. Released versions of the CDC-written TIPs do not use these ten field number/field value pairs.

These parameters are optional. If you specify a value, it must be within the following range:

$0 \leq fv9i \leq FF$

If you omit these parameters, there are no default values.

RTS PARAMETER

The RTS parameter specifies whether RTS input flow control is in effect on the line.

If this parameter is selected, CCP drops the RS-232C Request to Send (RTS) signal when it needs to regulate input flow on a line. When CCP can again receive input on the line, it will raise the RTS signal and the flow of input resumes.

RTS input flow control can be used to regulate input if and only if the following two conditions are satisified. The sending device (for example, the terminal connected to CCP) must be able to recognize one of the RS-232C signals as a sign to stop and start transmission of data. In addition, the connection between the NPU and the terminal must be wired such that RTS from CCP is received as the signal recognized by the terminal, most commonly Clear to Send (CTS).

This parameter is optional; it is valid for only asynchronous devices. If you specify the RTS parameter, you must use one of the following values:

NO   Indicates that the RTS signal is not raised or dropped to effect input flow control.

YES   Indicates that the RTS signal is raised and dropped to effect input flow control.

If you omit this parameter, the default is unknown and possibly undefined.

SDT PARAMETER

The SDT parameter specifies the subdevice type of the device you are defining. The subdevice type is the set of external characteristics of interest to the network software. This parameter is optional.

The following values are allowed definitions of a printer character set:

A6   Indicates that the device uses 64 ASCII characters.

A9   Indicates that the device uses 95 ASCII characters.

B6   Indicates that the device uses the 64-character CDC scientific (BCD) character set.

CCP   Indicates that your site-written TIP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

SDT12   Indicates a site-defined subdevice type that uses the subdevice type number 12.

SDT13   Indicates a site-defined subdevice type that uses the subdevice type number 13.

SDT14   Indicates a site-defined subdevice type that uses the subdevice type number 14.

SDT15   Indicates a site-defined subdevice type that uses the subdevice type number 15.

The following values are allowed definitions of a card reader punch pattern set:

CCP   Indicates that your site-written TIP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

26   Indicates that the device uses the 026 pattern set or the default pattern at the beginning of each job deck is that set.

29   Indicates that the device uses the 029 pattern set or the default pattern at the beginning of each job deck is that set.

The following values are allowed definitions of a plotter instruction byte size:

CCP   Indicates that your site-written TIP should use the default appropriate for the device; using this value is equivalent to omitting the parameter.

6BIT   Indicates that the device uses a 6-bit byte for each instruction.

8BIT   Indicates that the device uses an 8-bit byte for each instruction.

If you omit the SDT parameter or specify the value CCP, the default is unknown and possibly undefined.

STREAM PARAMETER

The STREAM parameter defines the stream number used within the terminal to address data to or from the device. This stream number should be unique for each device of the same type within the terminal. If the terminal has plotters, the stream number for each plotter should be unique among all plotters and card punches within the terminal.

This parameter is optional. If you specify this parameter, you must use one of the following values:

AUTOREC

Specifies that the stream number is to be determined by your site-written TIP when automatic recognition of the terminal occurs (valid only if the corresponding LINE or GROUP statement contains the AUTO parameter); using this value is equivalent to omitting the parameter.

$1 \leq streamno \leq 7$

Indicates the stream number associated with the device within the terminal.

If you omit this parameter or specify the value AUTOREC, the stream number determined during automatic recognition of the line is used.

TA PARAMETER

The TA parameter defines the terminal address of the device within the terminal device cluster.

The TA parameter is optional.

If you specify this parameter, you must use the following value:

AUTOREC

Indicates that your site-written TIP should determine the terminal address code during automatic recognition of the terminal (valid only if the AUTO parameter is used in the corresponding LINE or GROUP statement); using this value is equivalent to omitting the parameter.

If you omit this parameter or specify the value AUTOREC, the value determined by automatic recognition of the line is used.

UBL PARAMETER

The upline block limit (UBL) parameter specifies the number of upline blocks that can be outstanding (unacknowledged) between this terminal and an application program. The value you choose is independent of the ABL and DBL parameters.

You should choose a value that is larger than the maximum number of logical lines the terminal user can enter before output from the application program in the host must occur. A fixed number of input messages might have significance to an application program. If such an application does not perform output until receiving that number of input messages and if you choose a UBL value smaller than this number, the application program becomes dead-locked (it waits for blocks of data that the TIP might discard because the block limit has been reached for upline queuing). If the device must receive output before it can begin additional input, the device also becomes deadlocked by this situation.

This parameter is optional. The UBL parameter has the following range of values:

$1 \leq ubl \leq 31$

If you omit the UBL parameter, the default is unknown and possibly undefined.

UBZ PARAMETER

The upline block size (UBZ) parameter specifies the maximum number of character bytes each upline block can contain. The TIP probably divides each message from the terminal into blocks of the size you choose. The size you choose affects how much memory is needed to store the outstanding (unacknowledged) blocks of data in the NPU and the host.

You should choose a value that allows most of the logical lines entered by the device to fit into a single block so that messages need not be divided into multiple network blocks. You should also choose a value that can be efficiently manipulated by the host application program.

This parameter is optional. The UBZ parameter has the following range of values:

$$0 \leq ubz \leq 2043$$

If you specify 0, your site-written TIP should send an upline block whenever it receives 100 characters, or it detects a linefeed code, or when the page width is reached.

The value you declare should be chosen together with the value used for the UBL parameter. The NDL processor rounds the value you supply to the next multiple of 100 bytes, as follows:

Value Supplied	Value Used in File
0	0
$1 \leq ubz \leq 100$	100
$101 \leq ubz \leq 200$	200
.	.
.	.
.	.
$1901 \leq ubz \leq 2000$	2000

The NDL processor issues an informative messsage if it alters the value you specify.

For the Message Control System (MCS), the upline block size must be set to 0.

If you omit the UBZ parameter, the default is unknown and possibly undefined.

XBZ PARAMETER

The transmission block size (XBZ) parameter specifies the maximum number of character bytes each block sent to the terminal can contain. Your site-written TIP divides downline blocks as necessary to create a block of the specified number of characters. The value you choose should be less than or equal to the size of any buffer memory within the terminal.

This parameter is optional. The XBZ parameter has the following range of values:

$$200 \leq xbz \leq 2043$$

If you omit the XBZ parameter, the default is unknown and possibly undefined.

XLC PARAMETER

The XLC parameter indicates the maximum number of characters that can be input in each message of multiple-message transparent mode input from this device. After the terminal user or the application program changes the device's input mode to trans-

parent from normalized, the following occurs. When the device transmits the given number of characters, a message block is forwarded to the application and the device remains in transparent mode.

This parameter is optional. You can use the following values:

CCP

Indicates that the TIP should use the default number appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$$1 \leq xlc \leq 4095$$

Indicates the maximum decimal number of characters that the terminal can transmit as one multiple-message transparent mode message.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

XLTO PARAMETER

The XLTO parameter specifies whether a 200- to 400-millisecond timeout on the communication line ends multiple-message transparent mode input. After the terminal user or the application program changes the device's input mode to transparent from normalized, your site-written TIP probably forwards a message block to the application and changes back from transparent mode to normalized mode when the device stops transmitting for the indicated timeout period.

This parameter is optional. If you specify the XLTO parameter, you can use one of the following values:

CCP   Indicates that the TIP should use the default choice appropriate for the terminal class; using this value is equivalent to omitting the parameter.

NO   Indicates that a timeout does not end transparent mode input.

YES   Indicates that transparent mode input ends when a timeout occurs.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

XLX PARAMETER

The XLX parameter indicates the hexadecimal value of the character code that should cause upline transmission of a message in multiple-message transparent mode. If this character code is also used as the value for the XLY parameter and is input twice in succession, the device will be put back in normalized input mode as determined by the IN parameter.

This parameter is optional. The value that is valid for a given character depends on the character code set used by the terminal. You can use the following values:

CCP

Indicates that the TIP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \leq xlx \leq FF$

Indicates the hexadecimal code that ends each message within multiple-message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F might not be seen by the TIP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

XLY PARAMETER

The XLY parameter indicates the hexadecimal value of the character code that should end multiple-message transparent mode operation. After the terminal user or the application program changes the device's input mode to transparent from normalized, your site-written TIP probably changes back from transparent mode to normalized mode when the device transmits this character. This character code probably does not change the device's input mode if the code is also used as the value for the XLX parameter (unless it is input twice in succession).

You should select transparent mode input delimiters with care. The character code you declare as the XLY value must be the code of a character that the physical device can input. If the delimiter you declare is a character that cannot be input (either because of device hardware limitations or because of the PA parameter value declared for the device), then the Terminal Interface Program cannot terminate transparent mode input and the device will be trapped in that mode of operation once it has begun.

This parameter is optional. The value that is valid for a given character depends on the character code set used by the terminal. You can use one of the following values:

CCP

Indicates that the TIP should use the default character code appropriate for the terminal class; using this value is equivalent to omitting the parameter.

$0 \leq xly \leq FF$

Indicates the hexadecimal code that ends multiple-message transparent input.

Values above 7F should not be used unless the PA parameter value of N or I is used. Codes above 7F might not be seen by the TIP unless the eighth (parity) bit of each input byte is defined as data.

If you omit this parameter or specify the value CCP, the default is unknown and possibly undefined.

BUFFERING OF DATA

Sections 4 through 6 end with data buffering guidelines based on experience with CDC-written TIPs supporting CDC-defined devices in CDC-defined terminal classes. Those guidelines presume CDC-defined line types and CDC-written host application programs.

The local division describes those components of the network for which the Network Validation Facility in a host computer provides access control. These components are network application programs, device users, and directions for making and receiving application-to-application connections. These components are considered local to that host.

You can omit the local division from an NDL program if a new local configuration file is not needed. If no local configuration file exists for a host, you must create one, even if it is empty. There must be one LCF for each host in the network.

One local configuration file is created for each local division in your program. You can have more than one division so that you can create local configuration files for more than one host or alternate files for each host.

Only the local division of an NDL program applies to CDCNET networks.

LFILE STATEMENT

The LFILE statement names the local configuration file to be created. It must be the first statement, other than a COMMENT statement, in the local division. LFILE can be the only statement in the division if a file must be created but no local configuration requirements exist.

The NOS local file name specified in the LFILE statement should not be the same as that of another local file assigned to your job. The local configuration file is described in more detail in section 9.

The format of the LFILE statement is shown in figure 8-1. An example of a valid NFILE statement is

    LOCFILE: LFILE.

This statement creates a local configuration file with the NOS local file name LOCFILE.

```
lfile:  LFILE.

lfile   The local file name of the new local
        configuration file to be created.
        This name is required and must
        conform to NOS file name conven-
        tions; there is no default value.
```

Figure 8-1. LFILE Statement Format

USER STATEMENT

The USER statement can provide the login parameter values required by the Network Validation Facility (NVF) for a console and any associated batch devices when a terminal becomes active. This statement can also restrict access by a device to more than one network application program. The USER statement has the format shown in figure 8-2.

You can supply some or all of the login information (except a password value), so that the console user need not enter a value in response to each login prompt. NVF prompts the user for any values that you do not supply and omits prompts for values that you do supply. No prompt for password occurs when you supply a user name.

If you supply a mandatory value (MFAM, MUSER, or MAPPL parameter), the user is not prompted for an alternative. The user cannot override any supplied mandatory value and cannot specify a password.

If you supply a default value (DFAM, or DUSER parameter), the user is prompted for an alternative. The user can override any supplied default value and can specify a password.

If you specify a primary value (PFAM, PUSER, or PAPPL), the user is not prompted for an alternative. However, the user can override the supplied value.

The device names specified should match device names in the DEVICE or TERMDEV statements for the NCF being used by the network. When GROUP statements are used in the NCF, you must supply the NDL-generated device names. The NDL processor does not diagnose unmatched names; if no match exists, the USER statement has no effect.

The following USER statement restricts access to a single network application program, without login dialog:

    DEVA09: USER,MFAM=0,MUSER=NHP7600,MAPPL=IAF.

This statement causes all users of the console called DEVA09 to be automatically logged into IAF using the NOS system default family name and the user name NHP7600. No password validation occurs during this automatic login, and IAF is the only network application program that DEVA09 is permitted to access.

The following USER statement supplies unrestricted access to a single network application program:

    DEVA09: USER,DFAM=0,DUSER=NHP7600,PAPPL=IAF.

```
device:   USER [,MFAM=manafam,MUSER=manausr,MAPPL=manappl,DFAM=deftfam,DUSER=deftusr,PFAM=pfam,]

          [PUSER=puser,PAPPL=primapp].
```

device   The element name of the terminal device for which the information is being specified.
            This name should be an element name specified on or generated by NDL for a TERMDEV or
            DEVICE statement in the network division that creates the network configuration file for
            the network using this local configuration file.

manafam   The mandatory family name to use during automatic login of the device user. If MFAM is
            supplied, the user is never prompted for entry of a family name. If MFAM=0 is specified,
            the NOS system default family name is used; if MFAM is omitted or MFAM=NONE is specified,
            the user is always prompted for entry of a family name. Valid values for manafam are
            determined by each site. This parameter cannot be specified if the DFAM or PFAM parameter
            is specified.

manausr   The mandatory user name to use during automatic login of the device user. If MUSER is
            supplied, the user is never prompted for entry of a user name or password; if MUSER is
            omitted or MUSER=NONE is specified, the user is always prompted for entry of a user name
            and password. Valid values for manausr are determined by each site. This parameter
            cannot be specified if the DUSER or PUSER parameter is specified.

manappl   The mandatory application name to use during automatic login of the device user. If MAPPL
            is supplied, the user is never prompted for entry of an application program name; NVF
            connects the device to this program as soon as access validation is completed. If MAPPL
            is omitted or MAPPL=NONE is specified, the user is always prompted for entry of an
            application program name. The name must be that of an application program in the NOS
            common deck COMTNAP for the host using this local configuration file, or the value CS if
            the user is a network or diagnostic operator.

deftfam   The default family name to use during login of the device user. If DFAM is supplied, the
            user can enter a null line in response to the family name prompt. If DFAM=0 is specified,
            the NOS system default family name is used; if DFAM is omitted or DFAM=NONE is specified,
            the user must respond to the family name prompt with a valid, nonblank entry. The value
            specified can be overridden if the user enters a nonblank line. Valid values for
            deftfam are determined by each site. This parameter cannot be specified if the MFAM or
            PFAM parameter is specified.

deftusr   The default user name to use for the device during login. If deftusr is specified, the
            device user can enter a null line in response to the user name prompt; if such an entry
            occurs, the user is not prompted for a password; if DUSER is omitted or DUSER=NONE is
            specified, the device user must respond to the user name and password prompts with valid,
            nonblank entries. The value specified can be overridden if the user enters a nonblank
            line. This parameter cannot be specified if the MUSER or PUSER parameter is specified.

pfam   The primary auto-login family name. If PFAM is specified, the terminal user is not
            prompted for a family name on the first login attempt. This parameter cannot be specified
            if MFAM or DFAM is specified.

puser   The primary auto-login user name. If PUSER is specified, the terminal user is not
            prompted for a user name on the first login attempt. This parameter cannot be specified
            if MUSER or DUSER is specified.

primapp   The default (primary login) application name to use during initial login of the device
            user. If PAPPL is supplied, the user is not prompted for entry of an application program
            name during the first login attempt after connection to the host occurs; NVF connects the
            device to this program as soon as access validation is completed. If PAPPL is omitted or
            PAPPL=NONE is specified, the user must respond to the application program name prompt with
            a valid, nonblank entry. The name must be that of an application program in the NOS
            common deck COMTNAP for the host using this local configuration file, or the value CS if
            the user is a network or diagnostic operator. The value specified is overridden if the
            user enters a nonblank line.

Figure 8-2. USER Statement Format

This statement causes all users of the console called DEVA09 to be logged into IAF using the NOS system default family name and the user name NHP7600 by simply responding to each NVF prompt with an empty input line (usually generated by pressing the carriage return key). Password validation occurs during this login only if a user volunteers a value. IAF is the first but not the only network application program that DEVA09 is permitted to access.

The following USER statement configures a remote batch console and its related devices for access to RBF. The console is prompted only for family name and user name; the terminal operator can respond to both prompts with empty input lines. Connection to RBF is automatic:

        DEVA09: USER,DFAM=0,DUSER=NHP3456,PAPPL=RBF.

The network configuration defined for systems containing Centronics 533/536 and/or Hitachi 585 printers must include definitions for these printers. A USER statement is required to define each of these printers.

        devicex: USER MFAM=family,MUSER=PRINTxy,
                  MAPPL=PSU.

where xy is a number between 01 and 12, devicex is the name of the printer specified in the Network Configuration File (533/536 printers only), and family is the name of the family containing the user name PRINTxy.

PSU recognizes printers by the user name they log in with at the time they connect to PSU. Therefore, the printers must be defined under the user name PRINTxy to connect to the network. The user name PRINTxy must be validated for PSU.

APPL STATEMENT

The APPL statement defines special requirements for access to or execution of a network application program. APPL statements are not required for any CDC-written or site-written application program unless one of these requirements exists.

The format of the APPL statement is shown in figure 8-3. All APPL statement parameters apply to applications whether they are connected to CCP or CDCNET networks.

The CDC-written application programs are:

        Network Log Server (NETLS)

        Network Operator Facility (NOF)

        Network File Server (NETFS)

        Initialize MDI Server (INITMDI)

        Network Log Termination Utility (NLTERM)

        Interactive Facility (IAF)

        Interactive Transfer Facility (ITF)

        Message Control System (MCS)

        NOS/VE Interactive Facility (VEIAF)

        PLATO NAM Interface (PLATO)

        Printer Support Utility (PSU)

        Permanent File Transfer Initiator (PTF)

        Permanent File Transfer Server (PTFS)

        Queue File Transfer Initiator (QTF)

        Queue File Transfer Server (QTFS)

        Remote Batch Facility (RBF)

        Transaction Facility (TAF)

        Terminal Verification Facility (TVF)

Of these, the applications RBF, PTF, PTFS, QTF, QTFS, and TAF require special privileges as follows:

        RBF: APPL,UID,PRU.
        PTF: APPL,PRU,NETXFR,RS,MXCOPYS=3.
        PTFS: APPL,PRU,NETXFR,RS,MXCOPYS=15
        QTF: APPL,PRU,NETXFR,RS,MXCOPYS=3.
        QTFS: APPL,PRU,NETXFR,RS,MXCOPYS=15.
        TAF: APPL,UID.

These statements can be used regardless of whether the corresponding programs are installed in the host.

You could also define CDC-written programs with the statements:

        IAF: APPL,PRIV.
        MCS: APPL,PRIV.
        PLATO: APPL,PRIV.
        RBF: APPL,PRIV,UID,PRU.
        TAF: APPL,PRIV,UID.
        TVF: APPL.
        PTF: APPL,PRIV,PRU,NETXFR,RS,MXCOPYS=3.
        PTFS: APPL,PRIV,PRU,NETXFR,RS,MXCOPYS=15.
        QTF: APPL,PRIV,PRU,NETXFR,RS,MXCOPYS=3.
        QTFS: APPL,PRIV,PRU,NETXFR,RS,MXCOPYS=15.

These programs can execute without being defined as privileged, but this set of statements defines them that way to improve site security. Any program attempting to access the network (NETON) as one of these five programs but not meeting the entry point requirement is denied access. TVF must not have PRIV or PRIV=YES associated with it in an APPL statement.

The CDC-written application programs, NETLS, NOF, NETFS, INITMDI, and NLTERM are defined with the statements:

        NETLS: APPL,RS,PRIV.
        NOF: APPL,KDSP,RS,PRIV.
        NETFS: APPL,KDSP,RS,PRIV.
        INITMDI: APPL,KDSP.
        NLTERM: APPL,KDSP.

To avoid possible problems with the integrity of the network and with validated users' ability to access CDC-written network application programs, you should not allow use of the following names for site-written network application programs:

ALL	IAF	MLTF	NS	PTFS	TCF
BYE	INITMDI	NETFS	NUL	QTF	TVF
CS	ITF	NETLS	NVF	QTFS	VEIAF
DOP	LOGIN	NETOS	PFU	RBF	
FTS	LOGOUT	NETTU	PLATO	RMF	
FTFS	MCS	NLTERM	PSU	SCF	
HELLO	MHF	NOF	PTF	TAF	

service: APPL[,UID[=yn1],PRIV[=yn2],DI[=yn3],KDSP[=yn4],PRU[=yn5],NETXFR[=yn6],RS[=yn7],]

     [MXCOPYS=mxcopys].

service   The element name of the application program being configured as a network resource. This
          must be the name used by the program in its Network Access Method NETON statement and is the
          name terminal users must specify when logging in to it. If mxcopys = 1, this name can
          contain up to seven letters and digits starting with a letter. If mxcopys > 1, then this
          name can contain a maximum of five characters. This name cannot assume the following
          reserved application names: ALL, BYE, CS, HELLO, LOGIN, LOGOUT, NS, NUL, or NVF.

yn1   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this
          parameter indicates whether the application program requires users to have unique identi-
          fication. If UID or UID=YES is specified, only one terminal with a given user index and
          family name combination will be allowed connection with the application at a time. If UID
          is omitted or UID=NO is specified, more than one terminal at a time can be connected with
          the same combination of user index and family name.

yn2   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this param-
          eter identifies whether the application program must have privileged application status to
          access the network. If PRIV or PRIV=YES is specified, the program cannot connect to the
          network unless it contains an SSJ= entry point. If PRIV is omitted or PRIV=NO is specified,
          the program can access the network whether or not it contains an SSJ= entry point.

yn3   An optional reserved word value (YES or NO) for an optional stand-alone keyword; this param-
          eter specifies the status assigned to this program at network initiation. When DI or DI=YES
          is specified, the program is disabled at network initiation and cannot connect to the net-
          work until the HOP enables it. If DI is omitted or DI=NO is specified, the program is given
          an initial status of enabled unless the HOP specifies otherwise.

yn4   An optional reserved word value (YES or NO) which specifies whether the application program
          is allowed to use the NAM K-display. If KDSP or KDSP=YES is specified, the application is
          allowed to use the NAM K-display. If KDSP is omitted or KDSP=NO is specified, the
          application is not allowed to use the NAM K-display. KDSP is not supported for non-CDC
          applications.

yn5   An optional reserved word value (YES or NO) which specifies whether the application program
          is allowed to use the NAM PRU feature. If PRU or PRU=YES is specified, the application is
          allowed to use the NAM PRU feature. If PRU is omitted or PRU=NO is specified, the
          application is not allowed to use the NAM PRU feature. This parameter must be specified for
          the CDC supported applications RBF, QTF, PTF, QTFS, and PTFS. PRU is not supported for
          non-CDC applications.

yn6   An optional reserved word value (YES or NO) which specifies whether the application program
          is allowed to use the NAM file transfer capability NETXFR. If NETXFR or NETXFR=YES is
          specified, the application is allowed to use NETXFR. If NETXFR is omitted or NETXFR=NO is
          specified, the application is not allowed to use NETXFR. This parameter must be specified
          for the CDC supported applications QTF, PTF, QTFS, and PTFS. NETXFR is not supported for
          non-CDC applications.

yn7   An optional reserved word value (YES or NO) which specifies whether the application program
          is allowed to use the request startable capability. If the application is capable of
          supporting up to n number of connections (n is a parameter which the application program
          supplies to NAM), then if the RS parameter is specified, the first and each multiple of
          n+1st connection request to the application start up a new copy of the application, up to
          mxcopys number of applications. For each request startable application, a permanent file
          must be provided having the same user name that NAM is run under. The name of this file
          should be ZZxxxxx, where xxxxx is the first five characters of the application name.

mxcopys   The maximum number of copies $(1 \leq mxcopys \leq 15)$ of the application that can be
          simultaneously active on the host. If mxcopys is not specified, the default value of 1 is
          used. This parameter must be specified for the CDC-supported applications PTF, PTFS, QTF,
          and QTFS.

Figure 8-3. APPL Statement Format

Each valid user name in the NOS validation file (VALIDUs) has an internal identification value called a user index. Normally, each user name has a unique user index; however, it is possible for several user names to be assigned the same user index value.

The Network Validation Facility program checks the user index value currently associated with the terminal each time the terminal attempts connection to the network application program. If you used the unique identifier (UID) parameter for the program and another terminal with that index value is already connected to the program, the terminal requesting access receives a message indicating that the network application program is busy.

You should specify the UID parameter for RBF and TAF. These application programs route output files according to the combinations of user name and family name associated with the terminals they service. RBF and TAF output cannot be properly routed unless only one interactive terminal at a time with a given user name and family name combination is connected to the program.

If a network application program written by your site can support only one terminal with a given user name and family name login combination at a time, you should specify the UID parameter for it as well. Terminals capable of using the program can then use the same user name and family name combination for access to it without simultaneous access ever occurring.

If an application program should be available to terminal users only at certain times of the day, you can specify that it be initially disabled; the host operator can then enable it at the appropriate time. Unless an application program is enabled, it cannot access the network and a terminal cannot successfully log in to it. For example, suppose a program called TEST is configured by the statement

    TEST: APPL,DI.

TEST logically exists as a network resource within the host computer but cannot successfully access the network (NETON) or be accessed by a terminal user until the host operator enables it and it is initiated. You will probably want to configure your site-written application programs this way until they are thoroughly debugged.

Any application program defined in the local configuration file must also be identified with the same name in the operating system common deck COMTNAP. COMTNAP is used by NVF to validate terminal user login sequences. Bits in the access word for each user name correspond to site-written application programs in COMTNAP; other bits correspond to CDC-supplied application programs.

When NVF is given a program name during the login procedure, it searches COMTNAP for that name. The entry for the name tells which bit in the access word must be set for access to the program to be permitted. More than one program in COMTNAP can specify the same access word bit, so there is no formal limit on the number of application programs that are network resources. However, you must take care that the correct access word bit is set for all user names that should have permission to access a given application program.

For example, a network definition with separately named copies of the same application program is legal. In this case, the COMTNAP entries for all of the copies can specify the same access word bit. Each access word bit would then identify a superset or class of application programs, rather than a separate program.

COMTNAP and access word use are described in further detail in the NOS Version 2 Installation Handbook.

OUTCALL STATEMENT

The OUTCALL statement is specified for each path to each application that is to be used for application-to-application connection requests from application programs in the host being defined. The OUTCALL statement has the format shown in figure 8-4.

If more than one path exists between the calling host and the destination host, you should use multiple OUTCALL statements. Multiple OUTCALL statements should be specified in order according to the priority of paths.

OUTCALL statements are used to define call request data for calls between:

Two applications in a single host

Two applications in adjacent hosts sharing a front-end

Two applications in hosts connected by a trunk

Two applications in hosts separated by an X.25 PSN or X.25 line

The OUTCALL statement identifies the following information:

Addressing information used for calling an application program in another host or in the same host.

Validation information used by NVF to verify that the requesting application program is allowed to make an OUTCALL request.

Flow control information for the calling application and the network software to use on the application-to-application connection.

The optional user facilities to be used for the connection (for connections via X.25 links).

Optional call user data to be passed to a remote foreign host.

Physical identifiers for which the NOS LID/PID table needs to be dynamically updated by NAM.

The call user data is limited to 128 bytes by NHP. Four bytes are required for the protocol identifier which is always present. This leaves 124 bytes for the SERVICE, DOMAIN, and UDATA parameters. Parameter length validation for the above parameters is done by processing the parameters in the order, SERVICE, DOMAIN, and UDATA while maintaining a cumulative count of the number of bytes entered into the call user data field. If the value specified for a parameter results in the total call user data length exceeding 128 bytes, that parameter value is declared in error.

```
OUTCALL,NAME1=name1,NAME2/PID=name2/pidname[,NETOSD=path,SNODE=srcnode,DNODE=dstnode]

    [,ACCLEV=acclev,PRI=yn1,DBL=dwnblim,ABL=abl,DBZ=dwnlsiz]

    [,DHOST=dsthost,SHOST=srchost,PORT=portnum,WS=windsiz,DPLS=dpls]

    [,DTEA=dtea,UBL=upblim,UBZ=upbsize,FAC1=faccode,...,FAC31=faccode]

    [,PRID=protid,SERVICE=service,DOMAIN=domain,UDATA=udata]
```

name1   One of two keywords used to specify the OUTCALL block identifier. Name1 is required and can
        be from one to seven letters and digits only. A calling application program must specify
        both a name1 and a name2 to identify an OUTCALL block for use in establishing the connection
        to the called application program. Name 1 does not need to be specified for application
        programs that specify their own OUTCALL parameters. If the destination host application
        program is a CDC CYBER application program, the name1 parameter can specify the called
        application name. If name1 is not the application name on the CDC CYBER, the application
        name must be specified in the udata field.

name2   The second of two keywords used to specify the OUTCALL block identifier. Name2 is required
        (if PID is not specified) and can be from one to three letters and digits only. A calling
        application program must either explicitly or implicitly specify name2 unless an interhost
        application-to-application connection is required. This identifies an OUTCALL block for use
        in establishing the connection to the called application program. If the destination host
        application program is a CDC CYBER application program, the name2 parameter can specify the
        remote host identifier, which can be the remote host's machine identification.

pidname If this keyword is specified, it serves the same purpose as if name2 were specified. In
        addition, it causes NAM to indicate network service availability in the NOS LID/PID table.
        For the applications PTFx/QTFx, the PID form of the NAME2 parameter must be used. The PID
        form must also be used for an application that supplies its own OUTCALL parameters. When
        the PID is specified, it must also be specified in the LDCMid file for this host. Pidname
        can be from one to three letters and digits long.

path   The network types and destination host that compose the application-to-application path.
        Inclusion of this parameter allows NDL to determine appropriate default values for other
        OUTCALL parameters. You can specify any of the following values:

                        Originating   Destination   Destination
            Value   Network   Network   Operating System

            PPO   CCP   CCP   NOS
            PDO   CCP   CDCNET   NOS
            DPO   CDCNET   CCP   NOS
            DDO   CDCNET   CDCNET   NOS
            PDV   CCP   CDCNET   NOS/VE
            DDV   CDCNET   CDCNET   NOS/VE
            PAA   CCP   AOS/VS   AOS/VS
            DAA   CDCNET   AOS/VS   AOS/VS
            PFF   CCP   FOREIGN   FOREIGN
            DFF   CDCNET   FOREIGN   FOREIGN

        If path is not specified, the default value of PDO is used.

srcnode The decimal number (0 ≤ srcnode ≤ 255) specifying the source node of the logical link over
        which the application-to-application connection is made. The SNODE parameter should match
        the NODE value in a COUPLER statement used by the calling host as defined in the
        corresponding NCF. If srcnode is not specified, the default value of 0 is used for
        application-to-application connections within a single host.

dstnode The decimal node number (0 ≤ dstnode ≤ 255) specifying the destination node of the logical
        link over which the application-to-application connection is to be made. For
        application-to-application connections over a host-to-host logical link, this value is equal
        to the DHOST value and a coupler node number of the destination host. For
        application-to-application connections over a X.25 link, this value is equal to the node
        number of the NPU that accesses the PSN or direct X.25 link. If the pid form of defining
        name2 is used, a change of status for the logical link implied by snode and dnode causes a
        dynamic update of the logical identifiers (LIDs) associated with the corresponding physical
        identifier (PID). For application-to-application connections over X.25, the logical link
        terminates at the NPU so the status of the PID is governed by the status of this logical
        link regardless of the actual availability of the PID. If dstnode is not specified, the
        default value of 0 is used for application-to-application connections within the same host.

Figure 8-4. OUTCALL Statement Format (Sheet 1 of 4)

acclev   The decimal access level of the user (0 ≤ acclev ≤ 15) required to make an
           application-to-application connection using the OUTCALL definition. If acclev is not
           specified, any user or application program can make connections with the OUTCALL
           specification. If acclev is not specified, the default value of 0 is used. The user's
           access level is specified by using the AL parameter in the USER statement while creating the
           NOS validation file (see the NOS administration reference manual).

yn1   The stand-alone keyword value (YES or NO) for an optional stand-alone keyword. This
           parameter indicates whether data to the device is to have traffic priority over that to
           other devices in the network. Data to a device that does not have traffic priority is
           suspended to relieve network congestion before data to terminals with traffic priority is
           suspended. If PRI or PRI=YES is specified, the device has data traffic priority. If PRI is
           omitted or PRI=NO is specified, the device does not have data traffic priority.

dwnblim   The downline block limit. This parameter specifies the maximum number (1 ≤ dwnblim ≤ 7) of
           downline blocks that the calling host NAM is allowed to have outstanding (unacknowledged).
           If dwnblim is not specified, the default value of 2 is used.

abl   The application block limit. This parameter specifies the maximum number (1 ≤ abl ≤ 7) of
           blocks that the calling application program is allowed to have outstanding
           (unacknowledged). If abl is not specified, the default value of 2 is used.

dwnlsiz   The downline block size. This parameter specifies the maximum number (1 ≤ dwnlsiz ≤ 2043)
           of bytes that the calling application program can send in a block. If dwnlsiz is not
           specified, the default value of 255 is used.

dsthost   The address of the remote host (from 0 to FF) to which the application-to-application
           connection is being made. For intrahost or application-to-application connections over a
           host-to-host logical link, this value (if specified) should be equal to the DNODE value.
           For application-to-application connections over a X.25 link, this value is the hexadecimal
           address of the remote host or the destination host coupler for a CDC CYBER host. Dsthost is
           invalid when the destination operating system is NOS/VE (use the SERVICE parameter).
           Dsthost is required (if UDATA is not specified) when the destination operating system is NOS
           and the originating or destination network is CDCNET or the PORT parameter is specified;
           otherwise, the default value is the same as the DNODE value.

srchost   The logical identifier (1 to 6 hexadecimal digits) of the calling host. If the OUTCALL
           statement is to an application in a CDC CYBER/NAM host, it must be the ASCII code equivalent
           of the identifier you want passed to the receiving application. If srchost is not
           specified, the default is the three ASCII character equivalent of the SNODE value, right-
           justified and zero-filled.

portnum   The hexadecimal port number (1 ≤ portnum ≤ FE) on the interface NPU connected to the PSN or
           direct X.25 link over which the connection is to be made. This parameter has no meaning for
           trunk connections. If portnum is omitted, the default value of 0 is used.

windsiz   The window size to be used by CCP for this connection for the send direction. This
           parameter specifies the decimal number (1 ≤ windsiz ≤ 7) of outstanding packets allowed for
           a host-to-NPU logical link connection. The value specified for windsiz cannot exceed the
           maximum configured for the port. This parameter has no meaning for trunk connections. If
           windsiz is omitted, the default value of 2 is used.

dpls   The data packet length for the send direction. This parameter specifies the maximum number
           of data octets (8-bit bytes) that an X.25 packet for a called DTE can contain in the send
           direction. This parameter applies only to X.25 connections. The valid range of values for
           dpls is: 16, 32, 64, 128, 256, 512, and 1024. Values other than allowed values are
           translated to the next higher value, and values greater than 1024 cause an error message at
           NDL generation time. The value specified for dpls cannot exceed the maximum configured for
           the port. If dpls is omitted, the default value of 128 is used.

dtea   The PSN address of the remote PSN DTE interface with access to the destination host or
           application. This parameter is equivalent to the X.25 called DTE address. The valid range
           of values is 1 to 15 decimal digits. There is no default value. This parameter must be
           specified for X.25 application-to-application connections.

upblim   The upline block limit. This parameter specifies the maximum number (1 ≤ upblim ≤ 31) of
           blocks that the NPU can have outstanding (unacknowledged) to the calling host. This
           parameter is meaningful only for X.25 connections. If upblim is not specified, the default
           value of 2 is used.

Figure 8-4. OUTCALL Statement Format (Sheet 2 of 4)

upbsize   The upline block size divided by 100. This parameter specifies the maximum number $(1 \le$ upsize $\le 20)$ of bytes that the NPU can send to the calling host in a non-PRU data block. This parameter is only used for X.25 links. If upbsize is not specified, the default value of 2 is used.

faccode   The optional user group facility code and parameter definition. This parameter (of a possible set of parameters from FAC1 to FAC31) specifies the hexadecimal digit coding of a facility code and accompanying parameters. The range of valid values for each faccode is 4 to 12 hexadecimal digits. Any number of FAC parameters can be specified as long as the total number of hex digits is not greater than 63 octets (126 hexadecimal digits). The coding of the parameters depends on the type of PSN and/or destination facility definition and requirements. You should refer to the appropriate user facility definition before coding this parameter. Optional user facilities to specify the window size, data packet length, and throughput class must correspond to the WS and DPLS parameter settings, respectively. There is no default value; if faccode is not specified, no facilities are generated. However, some packet-switching networks, such as TELENET, insert facility codes by default in the call request packet sent to the destination mainframe. Contact your PSN vendor to see what, if any, facility codes will be inserted. If the PSN does insert facility codes, then these must be specified on a corresponding INCALL statement at the destination host. Otherwise, NVF will reject the incoming call because of a facilities mismatch between the sender and receiver.

protid   The protocol identification. This parameter specifies bytes 1 to 3 of the X.25 call user data field of the call request. This parameter tells the PSN or remote node of a direct X.25 link how the call user data is to be used. All X.25 connections to applications on remote hosts must have a PRID that signifies the call user data is to be transparent to the PSN and transferred as is. If the call user data is to be used for some other purpose by the PSN or remote receiving node, you should refer to the specific PSN specification for the exact coding of the PRID. Byte 4 of the protid is used internally by NAM and is set to zero when sent to the PSN. A maximum of 6 hexadecimal digits can be specified. The digits are encoded left-justified with zero-fill. If the parameter is not specified prior to the first UDATA definition, the following default hexadecimal values are used:

Destination Operating System	Value
NOS	C10000
NOS/VE	C20000
AOS/VS	C00000

The protid definition is intended to be used only for PSN logical link connections to foreign hosts. For connections between CDC CYBER mainframe applications, the default setting of the protocol identification provides sufficient information to establish the connection.

service   The title of an addressable service in the CDCNET network. The parameter value is defined as a LIST of elements of NOS/VE type NAME. NOS/VE type NAME is a string of 1 to 31 alphanumeric characters, including underscore, dollar, pound, and commercial at, _, $, #, @, respectively. Elements of the list are separated by a period. This is a required parameter when the destination operating system is NOS/VE. Service is required if the destination operating system is NOS/VE.

domain   The search domain. The parameter value is defined as a LIST of elements of NOS/VE type NAME. NOS/VE type NAME is a string of 1 to 31 alphanumeric characters, including underscore, dollar, pound, and commercial at, _, $, #, @, respectively. Elements of the list are separated by a period. If domain is not specified, the default value is a null domain field (empty string). The SERVICE parameter is required if DOMAIN is specified.

udata   The call user data parameter. This parameter specifies the hexadecimal string (1 to 256 hexadecimal digits) to be incorporated as bytes 5 through 128 of the call-user-data field of the OUTCALL request (the call request packet of X.25). The total number of digits, including the 8 digits of the protocol identifier, must not exceed 256. For an application-to-application connection using a PSN, the total number of digits must not exceed 32 (16 octets) unless the PSN supports the Fast Select facility. Even if the PSN supports the Fast Select facility, octets 17 through 128 are not passed to the destination host or application unless the destination subscribes to the Fast Select acceptance facility and the application supports the facility. To ease the burden for application programs requesting a connection to another application in a CDC CYBER host, the absence of the UDATA parameter invokes a default setting of the call user data to be used. If unspecified and the destination operating system is NOS or NOS/VE, the call-user-data section is formatted as:

Figure 8-4. OUTCALL Statement Format (Sheet 3 of 4)

```
        sssddaaaaaaa

            sss        The 3-ASCII character SHOST value, if specified, or the 3-ASCII character
                       string equivalent of the SNODE value, right-justified and zero-filled

            dd         The 2-ASCII character string equivalent of the DHOST value, right-justified
                       and zero-filled

            aaaaaaa    The 7-ASCII character string equivalent of the NAME1 value, left-justified and
                       blank character filled
```

For example, if SHOST=303031 (or SNODE=01 without an SHOST specification), DHOST=03, and
NAME1=APPLABC are defined without a UDATA specification, the call user data generated by
default would be the protocol identifier (C1000000$_{16}$), and SHOST(303031$_{16}$),
DHOST(3033$_{16}$) and NAME1 (4150504C414243$_{16}$). To make the OUTCALL request to another CDC
CYBER host resident application by means of a host-to-NPU logical link, the DHOST value
would have to specify the destination host coupler node number.

If unspecified and the destination operating system is AOS/VS, the call-user-data section is
formatted by taking the ASCII character equivalent of the NAME1 parameter and entering it as
the only call user data.

Figure 8-4. OUTCALL Statement Format (Sheet 4 of 4)

INCALL STATEMENT

The INCALL statement is specified for each possible
path to an application of the host being defined on
which application-to-application connections can be
received. The INCALL statement has the format shown
in figure 8-5.

When a connection request is received by a host,
NVF searches the LCF for a matching INCALL statement
based on the path from which the request came.
There can be several INCALL statements for a single
application. One reason for this is that there can
be several paths to access the host or application,
which necessitates different definitions regarding
physical addresses, validation, and the flow control
parameter. Another reason for this is that there
can be different facilities available to different
requesting applications or users.

INCALL statements are used to accept calls from an
application:

   In the same host

   In an adjacent host

   In a remote host connected by a trunk

   In a remote CDC CYBER 170 or foreign host con-
   nected by an X.25 PSN or X.25 direct link

The INCALL statement specifies the following infor-
mation:

   Addressing information from which requests are
   permitted.

   Flow control parameters to be used by the
   requested host.

The family or user name to be used by the
requesting application or user.

The name of the application program to which
this connection should be made.

The available facilities that are permitted for
connections using this INCALL statement.

Sites which use a CDCNET network to support their
terminal connections need supporting INCALL
statements. Specifying the INCALL statement for a
CDCNET network is required for internal system
usage.

The following INCALL statement parameters apply to
all CDCNET connections:

ABL	DBZ	SNODE	UNAME
ANAME	FAM	UBL	
DBL	SHOST	UBZ	

The following INCALL statement parameters apply
only to internal CDCNET network connections:

DPLS	PORT
FAC	WS

You should define the following INCALL statements
for the CDCNET applications NETLS, NOF, and NETFS:

   INCALL,FAM=0,UNAME=NETUSER,ANAME=NETLS,DBL=2,
      ABL=2,UBZ=20,DBZ=2000
   INCALL,FAM=0,UNAME=NETUSER,ANAME=NOF,DBL=2,
      ABL=2,UBZ=20,DBZ=2000
   INCALL,FAM=0,UNAME=NETUSER,ANAME=NETFS,DBL=7,
      ABL=7,UBZ=20,DBZ=2000

```
INCALL,FAM=famname,UNAME=usernam,ANAME=appname[,SNODE=srcnode]

   [,DNODE=dstnode,PRI=yn1,DBL=dwnblim,ABL=abl,DBZ=dwnlsiz]

   [,WS=sndsiz,WR=recsiz,DPLS=dpls,DPLR=dplr,SHOST=srchost,UBL=upblim,UBZ=upbsize]

   [,PORT=portnum,DTEA=dtea,COLLECT,FASTSEL,FAC1=faccode,...,FAC31=faccode]
```

famname   The family name under which the connection is validated. This name can be any valid one-
          through seven-character family name. This parameter is required and there is no default
          value. The use of this parameter does not depend on the type of network (CCP or CDCNET)
          used to reach the application.

usernam   The user name under which the connection is validated. This name can be any valid one-
          through seven-character user name and can contain letters, digits, and asterisks. This
          parameter is required and there is no default value. The use of this parameter does not
          depend on the type of network (CCP or CDCNET) used to reach the application.

appname   The requested application program name. This name must match the AAAAAAA field of the call
          user data of the call request packet. This parameter can be from one through seven ASCII
          characters left-justified and blank-filled. This parameter is required and there is no
          default value. The use of this parameter does not depend on the type of network used (CCP
          or CDCNET) used to reach the application.

srcnode   The decimal node number (0 $\leq$ srcnode $\leq$ 255) of the source node of the logical link over
          which the application-to-application connection request originated. For
          application-to-application connections over a host-to-host logical link, this is the coupler
          node number of the remote CDC CYBER host. For X.25 application-to-application connections,
          this is the node number of the NPU that interfaces to the PSN or direct X.25 link over which
          the application-to-application connection request was received. If srcnode is not
          specified, the default value of 0 is used. If the srcnode value specified is 0, the SNODE
          parameter is not used to match incoming requests to an INCALL block.

dstnode   The decimal number (0 $\leq$ dstnode $\leq$ 255) of the destination node of the logical link over
          which the application-to-application connection is received. If dstnode is not specified,
          the default value of 0 is used for application-to-application connections to the same host.
          If the dnode value specified is 0, the dnode parameter is not used to match incoming
          requests to an INCALL block. For CDCNET networks, this is the node number of the local
          mainframe interface.

yn1   The stand-alone keyword value (YES or NO) for an optional stand-alone keyword. This
          parameter indicates whether connection data from the calling host has traffic priority over
          data to other devices in the network. Data on a connection that does not have traffic
          priority is suspended to relieve network congestion before data on connections with traffic
          priority is suspended. If PRI or PRI=YES is specified, the connection has data traffic
          priority. If PRI is omitted or PRI=NO is specified, the connection does not have data
          traffic priority. This parameter is not meaningful for CDCNET networks.

dwnblim   The downline block limit. This parameter specifies the maximum number (1 $\leq$ dwnblim $\leq$ 31) of
          downline blocks that NAM in the called host can have outstanding (unacknowledged). If
          dwnblim is not specified, the default value of 2 is used. The use of this parameter does
          not depend on the type of network (CCP or CDCNET) used to reach the destination.

abl   The application block limit. This parameter specifies the maximum number (1 $\leq$ abl $\leq$ 7) of
          blocks that the called application program can have outstanding (unacknowledged). If abl is
          not specified, the default value of 2 is used. The use of this parameter does not depend on
          the type of network (CCP or CDCNET) used to reach the destination.

dwnlsiz   The downline block size. This parameter specifies the maximum number (1 $\leq$ dwnlsiz $\leq$ 2043)
          of bytes that the called application program can send in a block. If dwnlsiz is not
          specified, the default value of 255 is used. The use of this parameter does not depend on
          the type of network (CCP or CDCNET) used to reach the destination.

sndsiz   The window size to be used by CCP for the send direction for this connection. This
          parameter specifies the decimal value (1 $\leq$ sndsiz $\leq$ 7) of the number of outstanding packets
          allowed for the outbound direction of an X.25 connection. The value specified for sndsiz
          cannot exceed the maximum configured for the port. This parameter has no meaning for
          host-to-host logical link connections. If sndsiz is omitted, the default value of 2 is used.

Figure 8-5. INCALL Statement Format (Sheet 1 of 2)

recsiz   The window size to be used by CCP for the receive direction for this connection. This
          parameter specifies the decimal value (1 $\leq$ recsiz $\leq$ 7) of the number of outstanding packets
          allowed for the inbound direction of an X.25 connection. The value specified for recsiz
          cannot exceed the maximum configured for the port. This parameter has no meaning for
          host-to-host logical link connections. If recsiz is omitted, the default value of 2 is used.

dpls   The data packet length for the send direction. This parameter specifies the maximum number
          of data octets (8-bit bytes) that an X.25 packet to the called DTE can contain. The value
          specified for dpls cannot exceed the value configured for the port. If dpls is omitted, the
          default value of 128 is used. This parameter applies only to X.25 connections. The valid
          ranges of values for dpls is 16, 32, 64, 128, 256, 512, and 1024. Values other than
          allowed values are translated to the next higher value, and values greater than 1024 cause
          an error at NDL generation time.

dplr   The data packet length for the receive direction. This parameter specifies the maximum
          number of data octets (8-bit bytes) that an X.25 packet from the called DTE can contain.
          This parameter applies only to X.25 connections. The valid ranges of values for dplr is 16,
          32, 64, 128, 512, and 1024. Values other than the allowed values are translated to the next
          higher value, and values greater than 1024 cause an error at NDL generation time. The value
          specified for dplr cannot exceed the value configured for the port. If dplr is omitted, the
          default value of 128 is used.

srchost   The source host identifier (1 to 6 hexadecimal digits) of where the call request was
          originated. This value serves as the identifier of the calling host and must correspond to
          ASCII codes. If srchost is not specified, the default value is the ASCII string equivalent
          of 000. If srchost is set to 0 or not specified, this parameter is not used to match
          incoming requests to INCALL blocks. This parameter must be 0 for CDCNET connections.

upblim   The upline block limit. This parameter specifies the maximum number (1 $\leq$ upblim $\leq$ 7) of
          blocks that the NPU can have outstanding (unacknowleged) to the called host. This parameter
          is meaningful only to X.25 connections. If upblim is not specified, the default value of 2
          is used.

upbsize   The upline block size divided by 100. This parameter specifies the maximum number (1 $\leq$
          upbsize $\leq$ 20) of bytes that the local NPU can send in a non-PRU data block to the called
          host. If upbsize is not specified, the default value of 2 is used.

portnum   The hexadecimal port number (1 $\leq$ portnum $\leq$ FE) on the PSN access NPU through which the
          application-to-application connection request is received. This parameter is used for
          matching an incoming request to an INCALL block for only those requests received over an
          X.25 link. If this parameter is not specified, the check will not include this parameter.
          This parameter must be 0 for CDCNET connections.

dtea   The PSN address of the calling DTEA. The valid range of values is 1 to 15 decimal digits.
          This parameter is not applicable to application-to-application connections over host-to-host
          logical links. The default value of this parameter is 0. If DTEA is not specified, then
          DTEA is not used to match incoming call requests to the INCALL block.

COLLECT   The reverse charge facility indicator. If this parameter is specified, the reverse charge
          facility is supported for this connection and all charges for the connection are accepted.
          This parameter is applicable only for host-to-NPU logical link connections.

FASTSEL   The fast select facility indicator. If this parameter is specified, the fast select
          facility is supported for this connection. This parameter is applicable only for
          host-to-NPU logical link connections.

faccode   The facility code and parameter definition for a single facility field. This parameter (of
          a possible set of parameters from FAC1 to FAC31) specifies the hexadecimal digit coding of a
          facility and accompanying parameters. This parameter specifies the facilities (other than
          WR, WS, DPLR, DPLS, COLLECT, or FASTSEL) that are accepted on the incoming call request.
          The range of values for each faccode is 4 to 12 hexadecimal digits. There is no default
          value; if faccode is not specified, no facilities are accepted. This means that if the PSN
          has inserted facility codes by default, regardless of whether or not facility codes were
          declared on the OUTCALL statement, then NVF will reject the incoming call request because of
          facilities mismatch between sender and receiver. This parameter must be 0 for CDCNET
          connections.

Figure 8-5. INCALL Statement Format (Sheet 2 of 2)

This section describes the logical structure and content of the major NOS local files handled by the NDL processor. The NDL processor can create one or more additional scratch files during processing (local file names ZZZZZU1 through ZZZZZU5). These scratch files are not significant to you as an NDL programmer and are not described. Files not directly handled by the NDL processor, such as the NPU load file and VALIDUz, are described in the NOS Version 2 Installation Handbook and the NOS Version 2 Analysis Handbook.

The NDL processor handles four major types of local files:

The input file

Network configuration files

Local configuration files

The job listing file

The NDL processor also always produces a dayfile on file OUTPUT, indicating the error processing performed during execution. If no errors were encountered, a message indicating this is placed in the dayfile (see appendix B).

INPUT FILE

All file creation jobs (described in section 10) require an input file for the NDL processor. The input file contains all NDL statements required for the creation of the network definition files being generated (see section 1). The input file name is either INPUT, COMPILE, or a valid file name that you supply in the NDLP command. The input file is a sequential file of 80- or 90-character records, terminated by an end-of-record indicator. This file is processed using the operating system input/output macros.

NETWORK CONFIGURATION FILE

One network configuration file is created from each network division in the input file. This file contains information from the following NDL statements relating to the physical and logical configuration of elements:

NFILE	LINE
SUPLINK	GROUP
COUPLER	TERMINAL
NPU	TERMDEV
TRUNK	DEVICE
LOGLINK	

The network configuration file is constructed by the NDL processor. It is accessed by the Network Supervisor and the Communications Supervisor as a mass storage, direct access, random permanent file.

When the NDL processor finishes execution, any network configuration file it created remains attached to the job as a NOS local file. Postprocessing disposition of this file is your responsibility, using operating system commands. If one or more files without fatal errors were produced, you should make them direct access permanent files. These permanent files must have either file names known to the host operator or the default file names used in the procedure files that initiate network operation. Section 10 outlines the mechanism for naming the network configuration files; the operating system installation handbook describes the initiation procedure files.

When a network defined by a network configuration file is operating, the file is dedicated to that network. Another network configuration file cannot be substituted for the one being used, nor can the network configuration file in use be modified. The NDL processor can create a new network configuration file at any time, regardless of whether or not the network is operating. Once a network configuration file exists, it can be used the next time the network is initialized.

Each network configuration file contains records of several types. Only two types are of direct interest to you:

The header record, which contains file identification and verification information, and information from the TITLE statement.

The file index record, which contains a duplicate of the header record.

The file index record is the last record of a properly constructed file. This index record contains the validation information needed by the Network and Communication Supervisors to ensure that all NPUs are loaded according to the information in the same network configuration file. It also contains the system clock time when the file was created and the number and Programming System Report (PSR) level of the NDLP version that created the file. The time and date from the header record are not written as part of the page headers when the file is listed. The index record also contains a bit that indicates whether the file was created without fatal errors. If the NDL processor detects any fatal errors during creation of the file, it does not set that bit; the supervisory programs cannot use such a file, and the NDL processor cannot produce a file summary listing for it.

When the Network Supervisor accesses an old network configuration file at network initiation, it attempts to read the index record. If the index record does not meet network-wide compatibility criteria, the Network Supervisor requests another network configuration file from the host operator. The Network Supervisor also compares its code level with that found in the network configuration file index record. If the code levels are not the same, an informative message is sent to the host operator.

The NDL processor attempts to read the index record when it accesses an old network configuration file during a file inspection job (see section 10). If the index record does not have the validation bit set, the NDL processor issues a dayfile message (see appendix B). The processor will not read invalid files during file inspection jobs.

The formatted content of the other records is not significant to you. These records are described in the internal documentation of the network software.

LOCAL CONFIGURATION FILE

One local configuration file is created from each local division in the input file. It contains information from the following statements relating to the physical and logical configuration of the service elements of the network:

    APPL   INCALL   LFILE   OUTCALL   USER

The local configuration file is constructed by the NDL processor. It is accessed by the Network Validation Facility as a mass storage, direct access, random permanent file.

When the NDL processor finishes execution, any local configuration file it created remains attached to the job as a NOS local file. Postprocessing disposition of this file is the NDL programmer's responsibility, using operating system commands. If one or more files without fatal errors were produced, you should make them direct access permanent files. These permanent files must have file names known to the host operator or the default file names used in the procedure files that initiate network operation. Section 10 outlines the mechanism for naming the local configuration files; the operating system installation handbook describes the initiation procedure files.

When a network accessed according to a local configuration file is operating, the file is dedicated to that network. Another local configuration file cannot be substituted for the one being used, nor can the local configuration file in use be modified. The NDL processor can create a new local configuration file at any time, regardless of whether or not the network is operating. Once a local configuration file exists, it can be used the next time the network is initialized.

Each local configuration file contains records of several types. Only two of these are of direct interest to you:

The file header record, which contains information from the TITLE statement.

The validation record, which contains a duplicate of the header record and information similar to that in the network configuration file index record.

The validation record is the last record of a properly constructed file. This index record contains the system clock time when the file was created and the number and Programming System Report (PSR) level of the NDLP version that created the file. The time and date from the header record are not written as part of the page headers when the file is listed. The index record also contains a bit that indicates whether the file was created without fatal errors. If the NDL processor detects any fatal errors during creation of the file, it does not set that bit; the Network Validation Facility cannot use such a file, and the NDL processor cannot produce a file summary listing for it.

When the Network Validation Facility accesses an old local configuration file at network initiation, it attempts to read the index record. If the index record indicates that the file is invalid, the Network Validation Facility requests another local configuration file from the host operator.

When the NDL processor accesses an old local configuration file during a file inspection job, the processor attempts to read the validation record. If the validation record indicates that the file contains errors, the NDL processor issues a dayfile message and aborts the job. (See section 10 and appendix B.)

The formatted content of the other records is not significant to you. These records are described in the internal documentation of the network software.

JOB LISTING FILE

The NDL processor creates the job listing file by reading the other local files created. The job listing file contains any or all of the listing options described in section 10 and illustrated in section 11 and appendix B. The job listing file name is either OUTPUT or a NOS local file name that you have supplied in the NDLP command. If a job listing file name of zero is supplied, no job listing file is created.

This file is a sequential file of 137-character records, intended for output on a line printer. The file is created using the operating system input/output macros.

This section describes the logical structure of jobs using the NDL processor. The form such jobs have when they are input to the operating system for execution depends on whether the jobs are NDL programs creating new network definition files, or are commands intended to inspect existing network definition files.

All jobs using the NDL processor contain a command portion. File creation programs can contain an additional program portion as the input file (see section 9) and can contain data portions not related to NDL program processing. File inspection jobs do not contain a program portion but can contain data portions unrelated to NDL processor use.

COMMAND PORTION

The command portion of any job using the NDL processor begins with job name and NOS USER commands and can contain any command valid for the operating system. The commands required in this portion depend upon whether file creation or file inspection is performed.

FILE CREATION

The command portion of an NDL program job must contain the program execution command shown in figure 10-1. This program execution command causes the loading of the NDL processor from the system library.

The parameters in the NDLP program execution command are order-independent. If an option is desired, the keyword character and equals sign must be specified explicitly; a command of the following form is not valid:

    NDLP(TAPE1,TAPE2,SF)

The significance of parentheses and embedded blanks depends on the operating system conventions used. Commas are not used to positionally identify omitted parameters; a command of the following form, therefore, is also invalid:

    NDLP(,,LO=SF)

NDLP(I=ifn,L=ofn,LO=list)

ifn   The name of the local file containing the input source statements. If you omit the I parameter, the default name of INPUT is used. If you specify just the keyword character I but omit a file name, the default name of COMPILE is used.

ofn   The name of the local file to receive any listable output of the NDL program. This file name must meet operating system file name constraints. If you omit the L parameter, the default name of OUTPUT is used. If you specify L=0 (zero), no listings are created unless a fatal error is detected; if a fatal error is found, an error summary listing is written on file OUTPUT.

list   A list of one through four characters specifying the types of listing output you want the NDL processor to create. If you omit the LO parameter or specify just the keyword characters LO= , the default values indicated below are used. You can specify list values in any order. If you specify either default, the other default is suppressed. The list values can be:

     N   Indicates a normal source listing; N is a default value. This source listing contains all NDL statements as they were read from the input file. Each statement is prefixed with an NDL processor-generated sequence number; three asterisks precede a statement containing an error, and the letter D precedes a statement containing a defname identifier (see Special-Purpose Statements in section 2).

     D   Indicates a defname listing; this is a default value. This listing contains each defname and its associated value declarations from all DEFINE statements used in one division of the NDL program; the NDL processor creates separate listings for each division.

     S   Indicates an expanded source listing. This listing has the same format as a normal source listing; however, where a defname appears in the normal listing, the associated character string appears in the expanded listing.

     F   Indicates a file content summary listing. This listing consists of the title, type, and summarized content information for each of the validated network definition files produced by the NDL processor. Files containing fatal compilation errors are not listed.

Figure 10-1. NDLP File Creation Command Format

The following sample program execution commands are all valid:

NDLP.

Produces the defname and normal source listings by default.

NDLP(LO=SF)

Does not produce either default listing but produces both the expanded source listing and the file content summary listing.

NDLP(L=0)

Creates the network definition files but produces no listings. (If a fatal error is detected, the file OUTPUT contains a list of the diagnostic messages.)

NDLP(LO=NDSF)

Produces all four listings.

NDLP(I, LO=N)

Reads its input from the file called COMPILE and produces only the normal source listing; specification of only one of the listing default values always suppresses the other default.

The command portion should also contain a NOS DEFINE command to make the NOS local files created by the NDL processor direct access permanent files. This command must precede the NDLP command. The Network Supervisor, Communications Supervisor, and Network Validation Facility attach the network definition files using the permanent file names specified in the NAM startup procedure master file.

The DEFINE command should have the first form shown in figure 10-2 if the files specified are to be used as the network definition files attached by the released version of the NAM startup master file during network initialization. If these default permanent file names (NCFFILE and LCFFILE) are not used, any unique names can be specified as permanent file names for lfn1 and lfn2. When more than one network or local configuration file is created by the NDL program, only one set can be given the default permanent file names.

Any number of local or network configuration files can be made permanent files with appropriate alternative forms of the DEFINE command. Refer to the NOS 2 Reference Set, Volume 3, System Commands, for details.

The command portion can also contain an EXIT command and subsequent commands, if error exit processing of the job is desired. When NDL processor execution ends abnormally, job processing control transfers to any EXIT command present. The command portion is terminated by an EOR indicator, such as a multipunched card with 7/8/9 in column 1, or with /*EORnn if submitted through a multileaving station.

FILE INSPECTION

NDL processor file inspection is limited to producing file content summary listings for files

$$\text{DEFINE,} \quad \text{lfn1=} \begin{Bmatrix} \text{NCFFILE} \\ \text{pfn1} \end{Bmatrix}, \quad \text{lfn2=} \begin{Bmatrix} \text{LCFFILE} \\ \text{pfn2} \end{Bmatrix}.$$

lfn1   The local file name assigned to the network configuration file on the NFILE statement within the network division of the data portion of the job.

lfn2   The local file name assigned to the local configuration file on the LFILE statement within the local division of the data portion of the job.

pfn1   The permanent file name to assign to the network configuration file for use in the network initiation procedure files.

pfn2   The permanent file name to assign to the local configuration file for use in the network initiation procedure files.

Figure 10-2.   DEFINE Command Format

containing no fatal errors. File inspection jobs contain the form of the NDL processor command shown in figure 10-3. If the files to be inspected are not already local, the NDLP command must be preceded by a NOS ATTACH command valid for the two files concerned. The NDL processor can only list one local configuration file and one network configuration file for each NDLP command used. A file inspection job can be combined with a file creation job, but each portion requires a separate NDLP command of the appropriate type.

Figure 10-4 illustrates a possible combined inspection and creation run. The files are examined first, and file summaries are produced.

The parameters in the NDLP inspection command are order-independent. If an option is desired, the keyword character or characters must be specified explicitly; a statement of the following form is not valid:

NDLP(TAPE1,TSTLCF,TSTNCF)

The significance of parentheses and embedded blanks depends on the operating system conventions used. Commas are not used to positionally identify omitted parameters; a command of the following form is, therefore, also invalid:

NDLP(,,NF=TSTNCF)

If you omit all parameters from the parameter list, a dayfile error diagnostic can occur. The following command form is not valid for file inspection jobs:

NDLP.

```
NDLP (L=ofn,LF=Lfn1,NF=Lfn2)


ofn     The name of the local file to receive
        any listable output of the NDL pro-
        gram. This file name must meet
        operating system file name con-
        straints. If you omit the L param-
        eter, the default name of OUTPUT is
        used.

Lfn1    The local file name of the local con-
        figuration file for which you want a
        content summary listing. This value
        must be the name assigned to the per-
        manent file form of the local config-
        uration file in a preceding ATTACH
        statement or the local file name
        assigned in an LFILE statement in a
        preceding file creation portion of
        the job. This name is required;
        there is no default value. If you do
        not want a local configuration file
        content listing, omit the LF
        parameter.

Lfn2    The local file name of the network
        configuration file for which you want
        a content summary listing. This
        value must be the name assigned to
        the permanent file form of the net-
        work configuration file in a pre-
        ceding ATTACH statement or the local
        file name assigned in an NFILE state-
        ment in a preceding file creation
        portion of the job. This name is
        required; there is no default value.
        If you do not want a network config-
        uration file content listing, omit
        the NF parameter.
```

Figure 10-3. NDLP File Inspection
Command Format

because the NDL processor assumes that this state-
ment indicates a file creation job.

The following inspection control statements are all
valid:

    NDLP(LF=TSTLCF,NF=TSTNCF)

        Produces content summary listings of one
        local configuration file and one network
        configuration file.

    NDLP(LF=TSTLCF)

        Produces a content summary listing of one
        local configuration file.

    NDLP(LF=TSTLCF,L=TAPE1)

        Writes one local configuration file content
        summary listing on file TAPE1.

The command portion can contain an EXIT command and
subsequent commands, if you desire error exit
processing. When NDL processor execution ends
abnormally, job processing control transfers to any
EXIT command present.

The command portion is ended by a card with 7/8/9
multipunched in column 1, by a card with /*EORnn
punched in columns 1 through 7 if submitted through
a 2780, 3780, or HASP workstation, or by an EOR
indicator.

PROGRAM AND DATA PORTIONS

This portion of an NDL program's job structure
contains only the language statements described in
sections 3 and 4. The NDL processor makes no
provision for subroutines, subprograms, or proc-
esses written in compiler or assembly languages and
intended to be run as part of an NDL program.

```
Job File Contents:


ND98.
USER,USERNAM,PASSWOR.
CHARGE,CHARGENO,PROJECTNO.
ATTACH,PUBSNCF=NCFFILE,PUBSLCF=LCFFILE.
NDLP,LF=PUBSLCF,NF=PUBSNCF.◄─────────────Produces file summary listings
RETURN,*.
DEFINE,PUBSNCF,PUBSLCF.
ATTACH,COMPILE.
NDLP,I,LO=NDS.◄────────────────────────Creates file and source listings
End-of-record
End-of-information


COMPILE File Contents:


NDL source program statements from section 11
    .
    .
    .
End-of-record
End-of-information.
```

Figure 10-4. Example of Job for Both File Creation and File Inspection

Each program or data portion is ended by an EOR indicator, such as a multipunched card with 7/8/9 in column 1, or with /*EORnn in columns 1 through 7 if submitted through a 2780, 3780, or HASP workstation. If the job contains commands for postprocessing of the network definition files, subsequent data portions can be specified to provide directives.

The job is terminated by an end-of-information indicator, such as a card with 6/7/8/9 multipunched in column 1, or with /*EOI in columns 1 through 4 if submitted through a 2780, 3780, or HASP workstation. If the job contains only one program or data portion, you can omit the EOR indicator terminating that portion.

This section contains a debugged sample NDL file creation program and the listings produced when it is executed by the NDL processor. The dayfile listing is not shown.

PROGRAM INPUT

The input for this sample NDL program is submitted as a batch job under NOS. The job is structured to contain the command portion shown in figure 11-1 and a program portion (containing all of the NDL program statements listed in figure 11-2).

PROGRAM OUTPUT

The NDLP command in the input command portion of the job requests the NDL processor to produce all four possible types of output listing (figures 11-2 through 11-9). The normal source listing is shown in figures 11-2 and 11-6; the defname listing is shown in figures 11-3 and 11-7; the expanded source listing is given in figures 11-4 and 11-8; and the file content summary listing appears as figures 11-5 and 11-9. The time and date shown at the top of each listing page are the same as the time and date stored in the file header records of the network definition files produced by the program.

If errors are detected in the program, a multiple-page error summary is produced in the format described in appendix B. Dayfile messages are produced whether errors occur or not; these are also described in appendix B. Because the sample program contains no fatal errors, no error summary listing is shown.

```
ND99.
USER(LORI,PASWRD)
CHARGE(0059,7346219)
DEFINE(PUBSNCF=NCF,PUBSLCF=LCF)
NDLP(LO=F)
End-of-record
```

Figure 11-1. Sample Program Commands

If the DEFINE statement summary option (defname listing, chosen by LO=D) has been specified in the NDLP command but no DEFINE statements occur in the NDL program, a summary containing the following message is produced:

NO DEFINE COMMANDS ENCOUNTERED

A complete file creation job output listing with all options would be produced in the following order for each division:

Normal source listing

Error summary listing with columns to indicate line number, error code, error text, and keyword or value involved (only if errors occurred).

DEFINE statement summary (defname listing)

Expanded source listing

Network definition file summary (only if no fatal errors occurred).

NETWORK CONFIGURED

Figure 11-10 is a block diagram of the physical network configured by the NDL program shown in figures 11-1 through 11-9. The logical network configuration the program produces is shown in figure 11-11.

The NDL processor does not access the operating system equipment status table (EST). The EST entries in figure 11-10 are shown as background for interpreting the listing. The CCP build entries appear in the figure for the same reason; the NDL processor does not access or process the NPU load file in any manner.

No attempt is made in figure 11-10 to show the full parameterization of each configured terminal. Parameterization values other than defaults are indicated. Note that trunk TRNK2 is enabled in figure 11-10, while the corresponding logical link, LINK4, is disabled in figure 11-11. Shading in figure 11-11 shows the relationships between the local configuration file USER statements and the network configuration file DEVICE and TERMDEV statements.

```
LINE  ERR  DEFINE
  1               COMMENT   TEST CASE ND99.
  2               COMMENT   NAM 1 NDL REFERENCE MANUAL POSITIVE TEST SAMPLE, SECTION 11.
  3               COMMENT   NO ERRORS EXPECTED.
  4
  5               **********************************************************************
  6               *  THE FOLLOWING DIVISION CREATES A NETWORK DIVISION LOCAL FILE:
  7               **********************************************************************
  8
  9               PUBSNCF:  NFILE.
 10
 11                  TITLE,PUBSNCF (NDL REF MAN SAMPLE PROGRAM).
 12
 13               **********************************************************************
 14               *  NOTE THAT BLANK LINES ARE PERMITTED.
 15               **********************************************************************
 16
 17               **********************************************************************
 18               *  HERE ARE ALL OF THE DEFINE STATEMENTS TO BE EXPANDED FOR
 19               *  DEFINITIONS IN THIS DIVISION:
 20
 21               COMMENT   THIS DEFINES DEDICATED ASYNCHRONOUS AUTOMATICALLY RECOGNIZED
 22               *         COMMUNICATIONS LINES.
 23
 24               LDAAUTO:  DEFINE, LTYPE=A2, TIPTYPE=ASYNC, AUTO=YES.
 25
 26               COMMENT   THIS DEFINES DEDICATED ASYNCHRONOUS AUTOMATICALLY RECOGNIZED
 27               *         COMMUNICATION LINES FOR MODE 4 ONLY; IT IS NOT USED IN THIS
 28               *         NETWORK DEFINITION.
 29
 30               LDSAUTO:  DEFINE, LTYPE=S1, TIPTYPE=MODE4.
 31
 32               COMMENT   THIS DEFINES DIAL-UP ASYNCHRONOUS AUTOMATICALLY RECOGNIZED
 33               *         COMMUNICATION LINES.
 34
 35               LSAAUTO:  DEFINE, LTYPE=A1, TIPTYPE=ASYNC, AUTO=YES.
 36
 37               COMMENT   THIS DEFINES DIAL-UP SYNCHRONOUS AUTOMATICALLY RECOGNIZED
 38               *         COMMUNICATION LINES FOR MODE 4 ONLY; IT IS NOT USED IN THIS
 39               *         NETWORK DEFINITION.
 40
 41               LSSAUTO:  DEFINE, LTYPE=S2, TIPTYPE=MODE4.
 42
 43               **********************************************************************
 44               *  THIS 2550 IS CONFIGURED AS A FRONT-END NPU WITH A TERMINAL NODE
 45               *  OF 3.
 46
 47               NPA2550:  NPU, NODE=3, VARIANT=N2L, DMP=YES.
 48
 49               COMMENT   THERE IS ONE POSSIBLE SUPERVISION PATH FOR THIS NPU.  IT IS:
 50
 51                         SUPLINK, LLNAME=LINK1.
 52
 53               COMMENT   THIS NPU HAS ONE COUPLER THAT SERVES AS THE HOST END
 54               *         FOR TWO LOGICAL LINKS.  ONE LOGICAL LINK STARTS IN
 55               *         THIS NPU, THE OTHER STARTS IN A REMOTE NPU.
 56
 57               CPLR1:   COUPLER, NODE=1, HNAME=SYS173, LOC=PRIMARY.
 58                        LINK1:  LOGLINK, NCNAME=NPA2550.
 59                        LINK2:  LOGLINK, NCNAME=NPB2551.
 60
```

Figure 11-2. Normal Source Listing, Network Division (Sheet 1 of 6)

```
61          COMMENT    HERE ARE ALL OF THE COMMUNICATION LINES FOR THIS NPU:
62          *
63          *          THIS SYNCHRONOUS LINE SERVICES DIAL-UP MODE 4 WORK STATIONS
64          *          AND IS CONFIGURED FOR AUTOMATIC RECOGNITION.  AS CONFIGURED,
65          *          THE LINE CAN SERVICE HASP, 2780, 3780, OR MODE 4A TERMINALS
66          *          AND MODE 4C DEVICES, BUT ONLY MODE 4A AND 4C TERMINALS ARE
67          *          CONFIGURED FOR SUCCESSFUL AUTOMATIC RECOGNITION.
68
69          NPALN1:    LINE, PORT=2, LTYPE=S1, AUTO.
70
71                  TERMINAL, STIP=M4A, TC=200UT, CSET=BCD, CA=AUTOREC, RIC=NO.
72                      UT200A1:   DEVICE, DT=CON, TA=AUTOREC, HN=1, AUTOCON=YES,
73                                 DI=NO.
74                      UT200A2:   DEVICE, DT=CR, TA=AUTOREC, DI=NO.
75
76          COMMENT    NOTE THAT THE TERMINAL IS A BCD CODE SET, BUT THE PRINTER
77          *          HAS A 64-CHARACTER ASCII PRINT TRAIN (SDT):
78
79                      UT200A3:   DEVICE, DT=LP, SDT=A6, TA=AUTOREC, DI=NO.
80
81          COMMENT    EVEN THOUGH THIS TERMINAL IS REALLY A 734, IT IS CONFIGURED
82          *          AS IF IT WERE A 200 USER TERMINAL; TC DEFAULTS (CCP) TO TERM
83          *          CLASS USED FOR 200 USER TERMINALS.  THIS CONFIGURATION IS
84          *          AUTOMATICALLY RECOGNIZED AS DISTINCT FROM THAT OF THE
85          *          PREVIOUS DEFINITION ONLY BECAUSE OF ITS CHARACTER SET VALUE.
86
87                  TERMINAL, STIP=M4A, TC=CCP, CSET=ASCII, CA=AUTOREC.
88                      UT200B4:   DEVICE,  DT=CON, TA=AUTOREC, HN=1, AUTOCON=YES,
89                                 DI=NO.
90                      UT200B5:   DEVICE, DT=CR.
91                      UT200B6:   DEVICE, DT=LP, SDT=A6, TA=AUTOREC, DI=NO.
92
93          COMMENT    INSTEAD OF THE FOLLOWING TWO STATEMENTS, A SINGLE TERMDEV
94          *          STATEMENT COULD HAVE BEEN USED FOR THIS MODE 4C CONSOLE.
95
96                  TERMINAL, STIP=M4C, TC=CCP, CSET=ASCII, CA=AUTOREC.
97
98          COMMENT    DEFAULT DEVICE TYPE IS CONSOLE (CON); ADMINISTRATIVE OPERATOR
99          *          MESSAGES ARE LOCKED OUT (LK PARAMETER).
100
101                     BATCH3:  DEVICE, HN=1, AUTOCON, LK=YES.
102
103         COMMENT    THIS SYNCHRONOUS LINE SERVICES HASP WORKSTATIONS AND IS
104         *          CONFIGURED FOR AUTOMATIC RECOGNITION.  AS CONFIGURED, THE
105         *          LINE CAN SERVICE HASP, 2780, 3780, OR MODE 4A TERMINALS AND
106         *          MODE 4C DEVICES, BUT ONLY HASP DEVICES ARE CONFIGURED FOR
107         *          SUCCESSFUL AUTOMATIC RECOGNITION.
108
109         NPALN2:    LINE, PORT=3, LTYPE=S3, AUTO.
110
111         COMMENT    TC COULD BE OMITTED OR SPECIFIED AS TC=CCP FOR THE SAME
112         *          RESULT IN THE FOLLOWING:
113
114                  TERMINAL, STIP=PRE, TC=HPRE.
115                      HASPCON:   DEVICE, PRI=YES, HN=1, AUTOCON.
116                      HASPCP1:   DEVICE, DT=CP, DO=1.
117                      HASPPL1:   DEVICE, DT=PL, SDT=8BIT, DO=2.
118                      HASPCR1:   DEVICE, DT=CR, SDT=26, DO=1.
119                      HASPCR3:   DEVICE, DT=CR, SDT=26, DO=2.
120                      HASPLP1:   DEVICE, DT=LP, SDT=A9, DO=1.
121
```

Figure 11-2. Normal Source Listing, Network Division (Sheet 2 of 6)

```
122               COMMENT    THE FOLLOWING IS AN 029 CARD READER BY DEFAULT.
123
124                          HASPCR2:   DEVICE, DT=CR, DO=3.
125
126               COMMENT    THE FOLLOWING IS A 64-CHARACTER ASCII PRINTER BY DEFAULT.
127
128                          HASPLP2:   DEVICE, DT=LP, DO=2.
129
130               COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE CONFIGURED FOR
131               *          AUTOMATIC RECOGNITION; THE DEFINITION IS EXPANDED FROM A
132               *          DEFINE STATEMENT.
133
134       D       NPALN3:    LINE, PORT=4, LSAAUTO.
135
136               COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
137               *          DEVICE CAPABLE OF X-ON/X-OFF CONTROL OF INPUT OPERATIONS.
138
139                          HQRTTY1:   TERMDEV, STIP=N2741, TSPEED=110, IC=YES.
140
141               COMMENT    THE FOLLOWING IS A 2741-COMPATIBLE APL CHARACTER SET DEVICE.
142
143                          HQRTTY2:   TERMDEV, STIP=2741, TSPEED=134, CSET=EBCDAPL.
144
145               ***********************************************************************
146               *  THIS 2551 IS CONFIGURED AS A REMOTE NPU WITH A TERMINAL NODE OF 4.
147               *  IT REQUIRES AN OPERATOR COMMAND TO BEGIN SERVICING COMMUNICATION
148               *  LINES AFTER LOADING; IT IS NOT DUMPED BEFORE EACH LOAD (DMP=NO
149               *  IS THE DEFAULT).
150
151               NPB2551:   NPU, NODE=4, VARIANT=N5F, OPGO=YES.
152
153               COMMENT    THERE ARE TWO POSSIBLE SUPERVISION PATHS FOR THIS NPU.  IN ORDER
154               *          OF PRIORITY OF USE, THEY ARE:
155
156                          SUPLINK, LLNAME=LINK2.
157                          SUPLINK, LLNAME=LINK4.
158
159               COMMENT    THE FOLLOWING IS A DEDICATED ASYNCHRONOUS LINE CONFIGURED
160               *          FOR AUTOMATIC RECOGNITION; THE DEFINITION IS EXPANDED FROM
161               *          A DEFINE STATEMENT.
162
163       D       NPBLN1:    LINE, PORT=3, LDAAUTO.
164
165               COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER -COMPATIBLE
166               *          DEVICE, BUT IS CONFIGURED TO BE A HAZELTINE 2000.
167
168                          ARHTTY1:   TERMDEV, TC=H2000, TSPEED=AUTOREC.
169
170               COMMENT    THE FOLLOWING IS A 2741-COMPATIBLE DEVICE.
171
172                          ARHTTY2:   TERMDEV, STIP=2741, TSPEED=134, CSET=CORRES.
173
174               ***********************************************************************
175               *  THIS 2550 IS CONFIGURED AS A FRONT-END (LOCAL) NPU WITH A TERMINAL
176               *  MODE OF 5.  IT IS NOT DUMPED AFTER BEING LOADED.
177
178               NPC2550:   NPU, NODE=5, VARIANT=N3L, DMP=NO.
179
180               COMMENT    THERE IS ONE POSSIBLE SUPERVISON PATH FOR THIS NPU.  IT IS:
181
182                          SUPLINK, LLNAME=LINK3.
183
```

Figure 11-2. Normal Source Listing, Network Division (Sheet 3 of 6)

```
184              COMMENT    THIS NPU HAS ONE COUPLER THAT SERVES AS THE HOST END
185              *          FOR THREE LOGICAL LINKS.  ONE LOGICAL LINK STARTS IN THIS
186              *          NPU, THE OTHER TWO START IN REMOTE NPUS.
187
188              CPLR2:  COUPLER, NODE=2, HNAME=SYS173, LOC=PRIMARY.
189                      LINK3:  LOGLINK, NCNAME=NPC2550.
190                      LINK5:  LOGLINK, NCNAME=NPDAEG.
191                      LINK4:  LOGLINK, NCNAME=NPB2551, DI=YES.  *THIS LOGICAL LINK
192              *               CAN BE USED IF THE COUPLER CONNECTED TO NPA2550 IS DOWN.
193
194              COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN
195              *          AUTOMATIC RECOGNITION CONFIGURATION.
196
197     D        NPCLN1:  LINE, PORT=3, LSAAUTO.
198
199              COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
200              *          DEVICE, BUT IS CONFIGURED TO BE A MODEL 40 TELETYPE.
201
202              SVLTTY1:  TERMDEV, TC=M40, TSPEED=1200, HN=2, AUTOCON.
203
204              COMMENT    THE FOLLOWING IS A CDC 713 DEVICE.
205
206              SVLTTY2:  TERMDEV, TSPEED=300, TC=713, HN=2, AUTOCON.
207
208              **********************************************************************
209              *  THIS SELF-LOADING NPU IS CONFIGURED AS A REMOTE NPU WITH A TERMINAL
210              *  NODE OF 6.  AN OPERATOR COMMAND IS REQUIRED TO BEGIN SERVICING LINES
211              *  (YES IS IMPLICIT VALUE FOR THE OPGO PARAMETER).  THE DUMMY PICB NAME OF
212              *  "SELF" IS USED, AND DUMPING IS DISABLED BECAUSE RELOADING THROUGH A
213              *  NEIGHBOR NODE IS NOT POSSIBLE.
214
215              NPDAEG:  NPU, NODE=6, VARIANT=SELF, OPGO, DMP=NO.
216
217              COMMENT    THERE IS ONE POSSIBLE SUPERVISION PATH FOR THIS NPU.  IT IS:
218
219              SUPLINK, LLNAME=LINK5.
220
221              COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN AUTOMATIC
222              *          RECOGNITION CONFIGURATION.
223
224     D        NPDLN1:  LINE, PORT=3, LSAAUTO.
225
226              COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
227              *          DEVICE.
228
229              STATTY1:  TERMDEV, STIP=N2741, HN=2, AUTOCON=YES.
230
231              COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN AUTOMATIC
232              *          RECOGNITION CONFIGURATION.
233
234     D        NPDLN2:  LINE, PORT=2, LSAAUTO.
235
236              COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
237              *          DEVICE, BUT IS CONFIGURED TO BE A MODEL 37 TELETYPE.
238
239              STATTY2:  TERMDEV, TSPEED=150, TC=M33, PRI.
240
241              COMMENT    THE FOLLOWING IS A DEDICATED ASYNCHRONOUS LINE WITH AN
242              *          AUTOMATIC RECOGNITION CONFIGURATION.
243
244     D        NPDLN3:  LINE, PORT=4, LDAAUTO.
245
```

Figure 11-2. Normal Source Listing, Network Division (Sheet 4 of 6)

```
246      COMMENT   THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
247      *         DEVICE.
248
249                STATTY3:  TERMDEV, STIP=N2741, HN=2, AUTOCON=YES.
250
251      **********************************************************************
252      COMMENT   THIS 2550 IS CONFIGURED AS A FRONT-END WITH A TERMINAL NODE
253      *         OF 7.
254
255      NPE2550:  NPU, NODE=7, VARIANT=N7F, DMP=YES.
256
257      COMMENT   THERE ARE TWO POSSIBLE SUPERVISION PATHS FOR THIS NPU.
258      *         THEY ARE:
259
260                SUPLINK, LLNAME=LINK6.
261                SUPLINK, LLNAME=LINK8.
262
263      COMMENT   THIS NPU HAS TWO COUPLERS THAT SERVE AS THE HOST END FOR
264      *         THREE LOGICAL LINKS.
265
266      CPLR3:  COUPLER, NODE=8, HNAME=HOST3, LOC=PRIMARY.
267              LINK6:  LOGLINK, NCNAME=NPE2550.
268              LINK7:  LOGLINK, NCNAME=CPLR4.
269
270      CPLR4:  COUPLER, NODE=9, HNAME=SYS173, LOC=SECOND.
271              LINK8:  LOGLINK, NCNAME=NPE2550.
272
273      COMMENT   THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN
274      *         AUTOMATIC RECOGNITION CONFIGURATION.
275
276    D NPELN1:   LINE, PORT=2, LSAAUTO.
277
278      COMMENT   THE FOLLOWING IS A MODEL 40 TELETYPE.
279
280                TTY1: TERMDEV, TC=M40, TSPEED=1200, HN=8, AUTOCON.
281
282      COMMENT   THE FOLLOWING IS A COMMUNICATION LINE BETWEEN THE X.25
283      *         PACKET-SWITCHING NETWORK AND NPE2550.
284
285      NPELN2:   LINE, PORT=4, LTYPE=H1, TIPTYPE=X25, DFL=16, FRAME=1,
286                      RTIME=200, PSN=CDSN, NSVC=1.
287
288      COMMENT   THE FOLLOWING IS A CDC 713 DEVICE.
289
290                M713A: TERMDEV, STIP=PAD, TC=713.
291
292      **********************************************************************
293      *  THIS 2550 IS CONFIGURED AS A FRONT-END WITH A TERMINAL NODE OF 10.
294      *  THIS NPU IS CONNECTED BY A X.25 PACKET-SWITCHING NETWORK TO THE
295      *  NPU NPE2550.
296
297      NPF2550:  NPU, NODE=10, VARIANT=N6F, DMP=YES.
298
299      COMMENT   THERE IS ONE POSSIBLE SUPERVISION PATH FOR THIS NPU.  IT IS:
300
301                SUPLINK, LLNAME=LINK9.
302
303      COMMENT   THIS NPU HAS ONE COUPLER THAT SERVES AS THE HOST END FOR
304      *         ONE LOGICAL LINE WHICH STARTS IN THIS NPU.
305
306      CPLR5:  COUPLER, NODE=11, HNAME=HOST2, LOC=PRIMARY.
307              LINK9:  LOGLINK, NCNAME=NPF2550.
308
```

Figure 11-2. Normal Source Listing, Network Division (Sheet 5 of 6)

```
309          COMMENT    THE FOLLOWING IS A COMMUNICATION LINE BETWEEN THE X.25
310          *          PACKET-SWITCHING NETWORK AND THE NPU NPF2550.
311
312          NPFLN1:    LINE, PORT=3, LTYPE=H1, TIPTYPE=X25, DFL=16, FRAME=1,
313                        RTIME=200, PSN=CDSN, NSVC=1.
314
315          COMMENT    THE FOLLOWING IS A MODEL 40 TELETYPE.
316
317                     TTY2: TERMDEV, STIP=PAD, TC=M40.
318
319          ***********************************************************************
320          *  HERE ARE ALL OF THE TRUNK COMMUNICATION LINES IN THIS NETWORK.
321
322          COMMENT    THIS TRUNK CARRIES ONLY INTERACTIVE TRAFFIC, AND USES THE
323          *          DEFAULT FRAME SIZE OF 1050 CHARACTER BYTES:
324
325          TRNK1:     TRUNK, N1=NPA2550, P1=1, N2=NPB2551, P2=1, NOLOAD1=YES,
326                        NOLOAD2=YES, DI=NO.
327
328          COMMENT    NOTE THAT THIS TRUNK IS ENABLED, EVEN THOUGH THE CORRESPONDING
329          *          LOGICAL LINK IS DISABLED.
330
331          TRNK2:     TRUNK, N1=NPC2550, P1=1, N2=NPB2551, P2=2, NOLOAD1=YES,
332                        NOLOAD2=YES, FRAME=1050, DI=NO.
333
334          COMMENT    THIS SELF-LOADING NPU CANNOT BE LOADED BY ITS NEIGHBOR NODE
335          *          AND CANNOT LOAD ITS NEIGHBOR NODE.
336
337          TRNK3:     TRUNK, N1=NPC2550, P1=2, N2=NPDAEG, P2=1, NOLOAD1=YES,
338                        NOLOAD2=YES, FRAME=1050, DI=YES.
339
340          COMMENT    THIS IS THE END OF THE NETWORK DIVISION.
341
342          ***********************************************************************
343          *  THE FOLLOWING DIVISION CREATES A LOCAL DIVISION LOCAL FILE:
344          ***********************************************************************
345
```

Figure 11-2. Normal Source Listing, Network Division (Sheet 6 of 6)

```
               DEFINE NAME      DEFINE CONTENTS
               LDAAUTO          LTYPE=A2,TIPTYPE=ASYNC,AUTO=YES.
               LDSAUTO          LTYPE=S1,TIPTYPE=MODE4.
               LSAAUTO          LTYPE=A1,TIPTYPE=ASYNC,AUTO=YES.
               LSSAUTO          LTYPE=S2,TIPTYPE=MODE4.
```

Figure 11-3. DEFINE Statement Summary Listing,
Network Division

```
   LINE  ERR  DEFINE
      1              COMMENT    TEST CASE ND99.
      2              COMMENT    NAM 1 NDL REFERENCE MANUAL POSITIVE TEST SAMPLE, SECTION 11.
      3              COMMENT    NO ERRORS EXPECTED.
      4
      5              ***********************************************************************
      6              *  THE FOLLOWING DIVISION CREATES A NETWORK DIVISION LOCAL FILE:
      7              ***********************************************************************
      8
      9              PUBSNCF:  NFILE.
     10
     11                 TITLE,PUBSNCF (NDL REF MAN SAMPLE PROGRAM).
     12
     13              ***********************************************************************
     14              *  NOTE THAT BLANK LINES ARE PERMITTED.
     15              ***********************************************************************
     16
     17              ***********************************************************************
     18              *  HERE ARE ALL OF THE DEFINE STATEMENTS TO BE EXPANDED FOR
     19              *  DEFINITIONS IN THIS DIVISION:
     20
     21              COMMENT    THIS DEFINES DEDICATED ASYNCHRONOUS AUTOMATICALLY RECOGNIZED
     22              *          COMMUNICATIONS LINES.
     23
     24              LDAAUTO:  DEFINE, LTYPE=A2, TIPTYPE=ASYNC, AUTO=YES.
     25
     26              COMMENT    THIS DEFINES DEDICATED ASYNCHRONOUS AUTOMATICALLY RECOGNIZED
     27              *          COMMUNICATION LINES FOR MODE 4 ONLY; IT IS NOT USED IN THIS
     28              *          NETWORK DEFINITION.
     29
     30              LDSAUTO:  DEFINE, LTYPE=S1, TIPTYPE=MODE4.
     31
     32              COMMENT    THIS DEFINES DIAL-UP ASYNCHRONOUS AUTOMATICALLY RECOGNIZED
     33              *          COMMUNICATION LINES.
     34
     35              LSAAUTO:  DEFINE, LTYPE=A1, TIPTYPE=ASYNC, AUTO=YES.
     36
     37              COMMENT    THIS DEFINES DIAL-UP SYNCHRONOUS AUTOMATICALLY RECOGNIZED
     38              *          COMMUNICATION LINES FOR MODE 4 ONLY; IT IS NOT USED IN THIS
     39              *          NETWORK DEFINITION.
     40
     41              LSSAUTO:  DEFINE, LTYPE=S2, TIPTYPE=MODE4.
     42
     43              ***********************************************************************
     44              *  THIS 2550 IS CONFIGURED AS A FRONT-END NPU WITH A TERMINAL NODE
     45              *  OF 3.
     46
     47              NPA2550:  NPU, NODE=3, VARIANT=N2L, DMP=YES.
     48
     49              COMMENT    THERE IS ONE POSSIBLE SUPERVISION PATH FOR THIS NPU.  IT IS:
     50
     51                         SUPLINK, LLNAME=LINK1.
     52
     53              COMMENT    THIS NPU HAS ONE COUPLER THAT SERVES AS THE HOST END
     54              *          FOR TWO LOGICAL LINKS.  ONE LOGICAL LINK STARTS IN
     55              *          THIS NPU, THE OTHER STARTS IN A REMOTE NPU.
     56
     57              CPLR1:  COUPLER, NODE=1, HNAME=SYS173, LOC=PRIMARY.
     58                        LINK1:  LOGLINK, NCNAME=NPA2550.
     59                        LINK2:  LOGLINK, NCNAME=NPB2551.
     60
```

Figure 11-4. Expanded Source Listing, Network Division (Sheet 1 of 6)

```
61              COMMENT   HERE ARE ALL OF THE COMMUNICATION LINES FOR THIS NPU:
62              *
63              *         THIS SYNCHRONOUS LINE SERVICES DIAL-UP MODE 4 WORKSTATIONS
64              *         AND IS CONFIGURED FOR AUTOMATIC RECOGNITION.  AS CONFIGURED,
65              *         THE LINE CAN SERVICE HASP, 2780, 3780, OR MODE 4A TERMINALS
66              *         AND MODE 4C DEVICES, BUT ONLY MODE 4A AND 4C TERMINALS ARE
67              *         CONFIGURED FOR SUCCESSFUL AUTOMATIC RECOGNITION.
68
69              NPALN1:   LINE, PORT=2, LTYPE=S1, AUTO.
70
71                    TERMINAL, STIP=M4A, TC=200UT, CSET=BCD, CA=AUTOREC, RIC=NO.
72                        UT200A1:  DEVICE, DT=CON, TA=AUTOREC, HN=1, AUTOCON=YES,
73                                  DI=NO.
74                        UT200A2:  DEVICE, DT=CR, TA=AUTOREC, DI=NO.
75
76              COMMENT   NOTE THAT THE TERMINAL IS A BCD CODE SET, BUT THE PRINTER
77              *         HAS A 64-CHARACTER ASCII PRINT TRAIN (SDT):
78
79                        UT200A3:  DEVICE, DT=LP, SDT=A6, TA=AUTOREC, DI=NO.
80
81              COMMENT   EVEN THOUGH THIS TERMINAL IS REALLY A 734, IT IS CONFIGURED
82              *         AS IF IT WERE A 200 USER TERMINAL; TC DEFAULTS (CCP) TO TERM
83              *         CLASS USED FOR 200 USER TERMINALS.  THIS CONFIGURATION IS
84              *         AUTOMATICALLY RECOGNIZED AS DISTINCT FROM THAT OF THE
85              *         PREVIOUS DEFINITION ONLY BECAUSE OF ITS CHARACTER SET VALUE.
86
87                    TERMINAL, STIP=M4A, TC=CCP, CSET=ASCII, CA=AUTOREC.
88                        UT200B4:  DEVICE,  DT=CON, TA=AUTOREC, HN=1, AUTOCON=YES,
89                                  DI=NO.
90                        UT200B5:  DEVICE, DT=CR.
91                        UT200B6:  DEVICE, DT=LP, SDT=A6, TA=AUTOREC, DI=NO.
92
93              COMMENT   INSTEAD OF THE FOLLOWING TWO STATEMENTS, A SINGLE TERMDEV
94              *         STATEMENT COULD HAVE BEEN USED FOR THIS MODE 4C CONSOLE.
95
96                    TERMINAL, STIP=M4C, TC=CCP, CSET=ASCII, CA=AUTOREC.
97
98              COMMENT   DEFAULT DEVICE TYPE IS CONSOLE (CON); ADMINISTRATIVE OPERATOR
99              *         MESSAGES ARE LOCKED OUT (LK PARAMETER).
100
101                       BATCH3:  DEVICE, HN=1, AUTOCON, LK=YES.
102
103             COMMENT   THIS SYNCHRONOUS LINE SERVICES HASP WORKSTATIONS AND IS
104             *         CONFIGURED FOR AUTOMATIC RECOGNITION.  AS CONFIGURED, THE
105             *         LINE CAN SERVICE HASP, 2780, 3780, OR MODE 4A TERMINALS AND
106             *         MODE 4C DEVICES, BUT ONLY HASP DEVICES ARE CONFIGURED FOR
107             *         SUCCESSFUL AUTOMATIC RECOGNITION.
108
109             NPALN2:   LINE, PORT=3, LTYPE=S3, AUTO.
110
111             COMMENT   TC COULD BE OMITTED OR SPECIFIED AS TC=CCP FOR THE SAME
112             *         RESULT IN THE FOLLOWING:
113
114                   TERMINAL, STIP=PRE, TC=HPRE.
115                       HASPCON:  DEVICE, PRI=YES, HN=1, AUTOCON.
116                       HASPCP1:  DEVICE, DT=CP, DO=1.
117                       HASPPL1:  DEVICE, DT=PL, SDT=8BIT, DO=2.
118                       HASPCR1:  DEVICE, DT=CR, SDT=26, DO=1.
119                       HASPCR3:  DEVICE, DT=CR, SDT=26, DO=2.
120                       HASPLP1:  DEVICE, DT=LP, SDT=A9, DO=1.
121
```

Figure 11-4. Expanded Source Listing, Network Division (Sheet 2 of 6)

```
122          COMMENT   THE FOLLOWING IS AN 029 CARD READER BY DEFAULT.
123
124                    HASPCR2:  DEVICE, DT=CR, DO=3.
125
126          COMMENT   THE FOLLOWING IS A 64-CHARACTER ASCII PRINTER BY DEFAULT.
127
128                    HASPLP2:  DEVICE, DT=LP, DO=2.
129
130          COMMENT   THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE CONFIGURED FOR
131          *         AUTOMATIC RECOGNITION; THE DEFINITION IS EXPANDED FROM A
132          *         DEFINE STATEMENT.
133
134    D     NPALN3:   LINE, PORT=4, LTYPE=A1,TIPTYPE=ASYNC,AUTO=YES.
135
136          COMMENT   THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
137          *         DEVICE CAPABLE OF X-ON/X-OFF CONTROL OF INPUT OPERATIONS.
138
139                    HQRTTY1:  TERMDEV, STIP=N2741, TSPEED=110, IC=YES.
140
141          COMMENT   THE FOLLOWING IS A 2741-COMPATIBLE APL CHARACTER SET DEVICE.
142
143                    HQRTTY2:  TERMDEV, STIP=2741, TSPEED=134, CSET=EBCDAPL.
144
145          *********************************************************************
146          *  THIS 2551 IS CONFIGURED AS A REMOTE NPU WITH A TERMINAL NODE OF 4.
147          *  IT REQUIRES AN OPERATOR COMMAND TO BEGIN SERVICING COMMUNICATION
148          *  LINES AFTER LOADING; IT IS NOT DUMPED BEFORE EACH LOAD (DMP=NO
149          *  IS THE DEFAULT).
150
151          NPB2551:  NPU, NODE=4, VARIANT=N5F, OPGO=YES.
152
153          COMMENT   THERE ARE TWO POSSIBLE SUPERVISION PATHS FOR THIS NPU.  IN ORDER
154          *         OF PRIORITY OF USE, THEY ARE:
155
156                    SUPLINK, LLNAME=LINK2.
157                    SUPLINK, LLNAME=LINK4.
158
159          COMMENT   THE FOLLOWING IS A DEDICATED ASYNCHRONOUS LINE CONFIGURED
160          *         FOR AUTOMATIC RECOGNITION; THE DEFINITION IS EXPANDED FROM
161          *         A DEFINE STATEMENT.
162
163    D     NPBLN1:   LINE, PORT=3, LTYPE=A2,TIPTYPE=ASYNC,AUTO=YES.
164
165          COMMENT   THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
166          *         DEVICE, BUT IS CONFIGURED TO BE A HAZELTINE 2000.
167
168                    ARHTTY1:  TERMDEV, TC=H2000, TSPEED=AUTOREC.
169
170          COMMENT   THE FOLLOWING IS A 2741-COMPATIBLE DEVICE.
171
172                    ARHTTY2:  TERMDEV, STIP=2741, TSPEED=134, CSET=CORRES.
173
174          *********************************************************************
175          *  THIS 2550 IS CONFIGURED AS A FRONT-END (LOCAL) NPU WITH A TERMINAL
176          *  MODE OF 5.  IT IS NOT DUMPED AFTER BEING LOADED.
177
178          NPC2550:  NPU, NODE=5, VARIANT=N3L, DMP=NO.
179
180          COMMENT   THERE IS ONE POSSIBLE SUPERVISON PATH FOR THIS NPU.  IT IS:
181
182                    SUPLINK, LLNAME=LINK3.
183
```

Figure 11-4. Expanded Source Listing, Network Division (Sheet 3 of 6)

```
184              COMMENT    THIS NPU HAS ONE COUPLER THAT SERVES AS THE HOST END
185              *          FOR THREE LOGICAL LINKS.  ONE LOGICAL LINK STARTS IN THIS
186              *          NPU, THE OTHER TWO START IN REMOTE NPUS.
187
188              CPLR2:  COUPLER, NODE=2, HNAME=SYS173, LOC=PRIMARY.
189                      LINK3:  LOGLINK, NCNAME=NPC2550.
190                      LINK5:  LOGLINK, NCNAME=NPDAEG.
191                      LINK4:  LOGLINK, NCNAME=NPB2551, DI=YES.  *THIS LOGICAL LINK
192              *               CAN BE USED IF THE COUPLER CONNECTED TO NPA2550 IS DOWN.
193
194              COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN
195              *          AUTOMATIC RECOGNITION CONFIGURATION.
196
197        D     NPCLN1:  LINE, PORT=3, LTYPE=A1,TIPTYPE=ASYNC,AUTO=YES.
198
199              COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
200              *          DEVICE, BUT IS CONFIGURED TO BE A MODEL 40 TELETYPE.
201
202              SVLTTY1:  TERMDEV, TC=M40, TSPEED=1200, HN=2, AUTOCON.
203
204              COMMENT    THE FOLLOWING IS A CDC 713 DEVICE.
205
206              SVLTTY2:  TERMDEV, TSPEED=300, TC=713, HN=2, AUTOCON.
207
208              *******************************************************************
209              * THIS SELF-LOADING NPU IS CONFIGURED AS A REMOTE NPU WITH A TERMINAL
210              * NODE OF 6.  AN OPERATOR COMMAND IS REQUIRED TO BEGIN SERVICING LINES
211              * (YES IS IMPLICIT VALUE FOR THE OPGO PARAMETER).  THE DUMMY PICB NAME OF
212              * "SELF" IS USED, AND DUMPING IS DISABLED BECAUSE RELOADING THROUGH A
213              * NEIGHBOR NODE IS NOT POSSIBLE.
214
215              NPDAEG:  NPU, NODE=6, VARIANT=SELF, OPGO, DMP=NO.
216
217              COMMENT    THERE IS ONE POSSIBLE SUPERVISION PATH FOR THIS NPU.  IT IS:
218
219              SUPLINK, LLNAME=LINK5.
220
221              COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN AUTOMATIC
222              *          RECOGNITION CONFIGURATION.
223
224        D     NPDLN1:  LINE, PORT=3, LTYPE=A1,TIPTYPE=ASYNC,AUTO=YES.
225
226              COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
227              *          DEVICE.
228
229              STATTY1:  TERMDEV, STIP=N2741, HN=2, AUTOCON=YES.
230
231              COMMENT    THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN AUTOMATIC
232              *          RECOGNITION CONFIGURATION.
233
234        D     NPDLN2:  LINE, PORT=2, LTYPE=A1,TIPTYPE=ASYNC,AUTO=YES.
235
236              COMMENT    THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
237              *          DEVICE, BUT IS CONFIGURED TO BE A MODEL 37 TELETYPE.
238
239              STATTY2:  TERMDEV, TSPEED=150, TC=M33, PRI.
240
241              COMMENT    THE FOLLOWING IS A DEDICATED ASYNCHRONOUS LINE WITH AN
242              *          AUTOMATIC RECOGNITION CONFIGURATION.
243
244        D     NPDLN3:  LINE, PORT=4, LTYPE=A2,TIPTYPE=ASYNC,AUTO=YES.
245
```

Figure 11-4. Expanded Source Listing, Network Division (Sheet 4 of 6)

```
246            COMMENT   THE FOLLOWING TERMINAL COULD BE ANY TELETYPEWRITER-COMPATIBLE
247            *         DEVICE.
248
249                      STATTY3:  TERMDEV, STIP=N2741, HN=2, AUTOCON=YES.
250
251            ***********************************************************************
252            COMMENT   THIS 2550 IS CONFIGURED AS A FRONT-END WITH A TERMINAL NODE
253            *         OF 7.
254
255            NPE2550:  NPU, NODE=7, VARIANT=N7F, DMP=YES.
256
257            COMMENT   THERE ARE TWO POSSIBLE SUPERVISION PATHS FOR THIS NPU.
258            *         THEY ARE:
259
260                      SUPLINK, LLNAME=LINK6.
261                      SUPLINK, LLNAME=LINK8.
262
263            COMMENT   THIS NPU HAS TWO COUPLERS THAT SERVE AS THE HOST END FOR
264            *         THREE LOGICAL LINKS.
265
266            CPLR3:  COUPLER, NODE=8, HNAME=HOST3, LOC=PRIMARY.
267                    LINK6:  LOGLINK, NCNAME=NPE2550.
268                    LINK7:  LOGLINK, NCNAME=CPLR4.
269
270            CPLR4:  COUPLER, NODE=9, HNAME=SYS173, LOC=SECOND.
271                    LINK8:  LOGLINK, NCNAME=NPE2550.
272
273            COMMENT   THE FOLLOWING IS A DIAL-UP ASYNCHRONOUS LINE WITH AN
274            *         AUTOMATIC RECOGNITION CONFIGURATION.
275
276     D      NPELN1:   LINE, PORT=2, LTYPE=A1,TIPTYPE=ASYNC,AUTO=YES.
277
278            COMMENT   THE FOLLOWING IS A MODEL 40 TELETYPE.
279
280                      TTY1: TERMDEV, TC=M40, TSPEED=1200, HN=8, AUTOCON.
281
282            COMMENT   THE FOLLOWING IS A COMMUNICATION LINE BETWEEN THE X.25
283            *         PACKET-SWITCHING NETWORK AND NPE2550.
284
285            NPELN2:   LINE, PORT=4, LTYPE=H1, TIPTYPE=X25, DFL=16, FRAME=1,
286                            RTIME=200, PSN=CDSN, NSVC=1.
287
288            COMMENT   THE FOLLOWING IS A CDC 713 DEVICE.
289
290                      M713A: TERMDEV, STIP=PAD, TC=713.
291
292            ***********************************************************************
293            *  THIS 2550 IS CONFIGURED AS A FRONT-END WITH A TERMINAL NODE OF 10.
294            *  THIS NPU IS CONNECTED BY A X.25 PACKET-SWITCHING NETWORK TO THE
295            *  NPU NPE2550.
296
297            NPF2550:  NPU, NODE=10, VARIANT=N6F, DMP=YES.
298
299            COMMENT   THERE IS ONE POSSIBLE SUPERVISION PATH FOR THIS NPU.  IT IS:
300
301                      SUPLINK, LLNAME=LINK9.
302
303            COMMENT   THIS NPU HAS ONE COUPLER THAT SERVES AS THE HOST END FOR
304            *         ONE LOGICAL LINE WHICH STARTS IN THIS NPU.
305
306            CPLR5:  COUPLER, NODE=11, HNAME=HOST2, LOC=PRIMARY.
307                    LINK9:  LOGLINK, NCNAME=NPF2550.
308
```

Figure 11-4. Expanded Source Listing, Network Division (Sheet 5 of 6)

```
309          COMMENT   THE FOLLOWING IS A COMMUNICATION LINE BETWEEN THE X.25
310          *         PACKET-SWITCHING NETWORK AND THE NPU NPF2550.
311
312          NPFLN1:   LINE, PORT=3, LTYPE=H1, TIPTYPE=X25, DFL=16, FRAME=1,
313                      RTIME=200, PSN=CDSN, NSVC=1.
314
315          COMMENT   THE FOLLOWING IS A MODEL 40 TELETYPE.
316
317                    TTY2: TERMDEV, STIP=PAD, TC=M40.
318
319          ************************************************************************
320          * HERE ARE ALL OF THE TRUNK COMMUNICATION LINES IN THIS NETWORK.
321
322          COMMENT   THIS TRUNK CARRIES ONLY INTERACTIVE TRAFFIC, AND USES THE
323          *         DEFAULT FRAME SIZE OF 1050 CHARACTER BYTES:
324
325          TRNK1:  TRUNK, N1=NPA2550, P1=1, N2=NPB2551, P2=1, NOLOAD1=YES,
326                    NOLOAD2=YES, DI=NO.
327
328          COMMENT   NOTE THAT THIS TRUNK IS ENABLED, EVEN THOUGH THE CORRESPONDING
329          *         LOGICAL LINK IS DISABLED.
330
331          TRNK2:  TRUNK, N1=NPC2550, P1=1, N2=NPB2551, P2=2, NOLOAD1=YES,
332                    NOLOAD2=YES, FRAME=1050, DI=NO.
333
334          COMMENT   THIS SELF-LOADING NPU CANNOT BE LOADED BY ITS NEIGHBOR NODE
335          *         AND CANNOT LOAD ITS NEIGHBOR NODE.
336
337          TRNK3:  TRUNK, N1=NPC2550, P1=2, N2=NPDAEG, P2=1, NOLOAD1=YES,
338                    NOLOAD2=YES, FRAME=1050, DI=YES.
339
340          COMMENT   THIS IS THE END OF THE NETWORK DIVISION.
341
342          ************************************************************************
343          * THE FOLLOWING DIVISION CREATES A LOCAL DIVISION LOCAL FILE:
344          ************************************************************************
345
```

Figure 11-4. Expanded Source Listing, Network Division (Sheet 6 of 6)

```
        VALID NCF CREATED      06.48.28  85/08/07

NCF FILE NAME          PUBSNCF

NPU          NAME     NODE      VARIANT   OPGO      DMP
             NPA2550   3        N2L       NO        YES

   SUPLINK   LLNAME
             LINK1

   TRUNK     NAME     N1        N2   P1 P2   NOLOAD1  NOLOAD2  STATUS    FRAME
             TRNK1    NPA2550   NPB2551  1  1     YES      YES      EN        1050

   COUPLER   NAME     NODE      HNAME     LOC
             CPLR1     1        SYS173    PRIMARY

   LOGLINK   NAME     NCNAME    STATUS
             LINK1    NPA2550   EN
             LINK2    NPB2551   EN

   LINE      NAME     PORT/ LTYPE   AUTO/ TIPTYPE/ DI   LSPEED/  DFL/  FRAME/ RTIME/ RCOUNT/ NSVC/
                      LCN           IMDISC  RC          XAUTO    PSN   NPVC   AL     ARSPEED  DTEA

             NPALN1    2    S1      YES            NO
                       0            NO                  NO                     0    NO

   TERMINAL  STIP/ TC     CSET     TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
             PAD
             M4A   200UT  BCD              0   NO                                      NO

   DEVICE    NAME     DT/   SDT/  ABL/  DBZ/  UBL/     HN/  AUTOCON/  BR/   AB/  B1/  CI/  CT/  DLC/   EP/  LI/  PG/  PL/  SE/
                      TA    XBZ/  DBL/  UBZ/  STREAM/  DO/     HD     PRI/  BS/  B2/  CN/  DLX/ DLTO/  IN/  OP/  PA/  PW/  STAT/
                            MCI   MLI   RTS   XLY      MC              FA   ELO  ELX  ELR  EBO  EBX    EBR  CP   IC   OC   LK

             UT200A1  CON         2     1040  7        1    YES
                       0   1040   2     100               NO        NO                                                    EN

             UT200A2  CR                      7             NO
                       0   1040   2     640            1              NO                                                  EN

             UT200A3  LP    A6          640   7        1    NO
                       0   1040   1                    1              NO                                             64   EN
```

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 1 of 9)

```
TERMINAL  STIP/  TC    CSET    TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
          PAD
          M4A          ASCII            0  NO                                    NO

DEVICE    NAME   DT/  SDT/  ABL/  DBZ/  UBL/   HN/  AUTOCON/ BR/   AB/  B1/  CI/ CT/ DLC/   EP/ LI/ PG/ PL/ SE/
                 TA   XBZ/  DBL/  UBZ/  STREAM/ DO/      HD  PRI/  BS/  B2/  CN/ DLX/ DLTO/  IN/ OP/ PA/ PW/ STAT/
                      MCI   MLI   RTS   XLY     MC           FA    ELO  ELX  ELR EBO  EBX    EBR CP  IC  OC  LK

          UT200B4  CON      2   1040    7      1     YES
                   0  1040  2    100                 NO       NO                                               EN

          UT200B5  CR               7            NO
                   0  1040  2   640         1                 NO                                               EN

          UT200B6  LP  A6      640    7            NO                                                   64
                   0  1040  1          1                      NO                                               EN

TERMINAL  STIP/  TC    CSET    TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
          PAD
          M4C          ASCII            0  NO                                    NO

DEVICE    NAME   DT/  SDT/  ABL/  DBZ/  UBL/   HN/  AUTOCON/ BR/   AB/  B1/  CI/ CT/ DLC/   EP/ LI/ PG/ PL/ SE/
                 TA   XBZ/  DBL/  UBZ/  STREAM/ DO/      HD  PRI/  BS/  B2/  CN/ DLX/ DLTO/  IN/ OP/ PA/ PW/ STAT/
                      MCI   MLI   RTS   XLY     MC           FA    ELO  ELX  ELR EBO  EBX    EBR CP  IC  OC  LK

          BATCH3   CON      2   1280    7      1     YES
                   0  1280  2    100                 NO       NO                                               EN
                                                                                                              YES

LINE      NAME   PORT/ LTYPE  AUTO/ TIPTYPE/ DI   LSPEED/  DFL/  FRAME/ RTIME/ RCOUNT/ NSVC/
                 LCN          IMDISC  RC          XAUTO    PSN   NPVC   AL    ARSPEED  DTEA

          NPALN2  3    S3     YES           NO
                  0           NO                  NO                    0     NO

TERMINAL  STIP/  TC    CSET    TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
          PAD
          PRE    HPRE                    YES
```

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 2 of 9)

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	HASPCON	CON		2	400	7	1	YES											
			400	2	100	0		NO	YES										EN
	HASPCP1	CP			640	7		NO											
			400	1		0	1		NO										EN
	HASPPL1	PL	8BIT		640	7		NO											
			400	1		0	2		NO										EN
	HASPCR1	CR	26			7		NO											
			400	2	640	0	1		NO										EN
	HASPCR3	CR	26			7		NO											
			400	2	640	0	2		NO										EN
	HASPLP1	LP	A9		640	7		NO										64	
			400	1		0	1		NO										EN
	HASPCR2	CR				7		NO											
			400	2	640	0	3		NO										EN
	HASPLP2	LP			640	7		NO										64	
			400	1		0	2		NO										EN

LINE	NAME	PORT/ LCN	LTYPE	AUTO/ IMDISC	TIPTYPE/ RC	DI	LSPEED/ XAUTO	DFL/ PSN	FRAME/ NPVC	RTIME/ AL	RCOUNT/ ARSPEED	NSVC/ DTEA
	NPALN3	4	A1	YES	ASYNC	NO						
		0		NO			NO			0	YES	

TERMINAL	STIP/ PAD	TC	CSET	TSPEED	CA	RIC	CO	BCF	MREC	W	CTYP	NCIR	NEN	EOF	COLLECT
	N2741			110		NO									

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	HQRTTY1	CON		1	230	7	1	YES											EN
			230	1	100			NO	NO									YES	

TERMINAL	STIP/ PAD	TC	CSET	TSPEED	CA	RIC	CO	BCF	MREC	W	CTYP	NCIR	NEN	EOF	COLLECT
	2741		EBCDAPL	134		NO									

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 3 of 9)

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	HQRTTY2	CON	230	1 1	230 100	7	1	YES NO	NO										EN

TIPTYPES USED FOR THIS NPU

ASYNC   SYNAUTO

NPU	NAME	NODE	VARIANT	OPGO	DMP
	NPB2551	4	N5F	YES	YES

SUPLINK	LLNAME
	LINK2
	LINK4

TRUNK	NAME	N1	N2	P1	P2	NOLOAD1	NOLOAD2	STATUS	FRAME
	TRNK1	NPA2550	NPB2551	1	1	YES	YES	EN	1050
	TRNK2	NPC2550	NPB2551	1	2	YES	YES	EN	1050

LINE	NAME	PORT/ LCN	LTYPE	AUTO/ IMDISC	TIPTYPE/ RC	DI	LSPEED/ XAUTO	DFL/ PSN	FRAME/ NPVC	RTIME/ AL	RCOUNT/ ARSPEED	NSVC/ DTEA
	NPBLN1	3 0	A2	YES NO	ASYNC	NO	NO			0	YES	

TERMINAL	STIP/ PAD	TC	CSET	TSPEED	CA	RIC	CO	BCF	MREC	W	CTYP	NCIR	NEN	EOF	COLLECT
	N2741	H2000				NO									

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	ARHTTY1	CON	230	1 1	230 100	7	2	YES NO	NO										EN

TERMINAL	STIP/ PAD	TC	CSET	TSPEED	CA	RIC	CO	BCF	MREC	W	CTYP	NCIR	NEN	EOF	COLLECT
	2741		CORRES	134		NO									

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	ARHTTY2	CON	230	1 1	230 100	7	2	YES NO	NO										EN

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 4 of 9)

TIPTYPES USED FOR THIS NPU

ASYNC

```
NPU          NAME       NODE     VARIANT   OPGO      DMP
             NPC2550    5        N3L       NO        NO

SUPLINK   LLNAME
          LINK3

TRUNK     NAME       N1        N2       P1 P2   NOLOAD1  NOLOAD2  STATUS   FRAME
          TRNK2      NPC2550   NPB2551  1  2    YES      YES      EN       1050
          TRNK3      NPC2550   NPDAEG   2  1    YES      YES      DI       1050

COUPLER   NAME       NODE      HNAME     LOC
          CPLR2      2         SYS173    PRIMARY

LOGLINK   NAME       NCNAME    STATUS
          LINK3      NPC2550   EN
          LINK4      NPB2551   DI
          LINK5      NPDAEG    EN

LINE       NAME      PORT/ LTYPE  AUTO/ TIPTYPE/ DI   LSPEED/  DFL/   FRAME/  RTIME/ RCOUNT/ NSVC/
                     LCN          IMDISC RC          XAUTO    PSN    NPVC    AL     ARSPEED  DTEA

           NPCLN1    3     A1     YES   ASYNC    NO
                     0            NO                   NO                     0      YES
TERMINAL   STIP/ TC        CSET    TSPEED CA RIC CO    BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
           PAD
           N2741 M40               1200       NO

DEVICE    NAME       DT/   SDT/  ABL/  DBZ/   UBL/     HN/  AUTOCON/ BR/   AB/  B1/  CI/ CT/ DLC/   EP/  LI/  PG/  PL/ SE/
                     TA    XBZ/  DBL/  UBZ/   STREAM/  DO/     HD    PRI/  BS/  B2/  CN/ DLX/ DLTO/  IN/  OP/  PA/  PW/ STAT/
                           MCI   MLI   RTS    XLY      MC            FA    ELO  ELX  ELR EBO  EBX    EBR  CP   IC   OC  LK

          SVLTTY1    CON         1     230    7        2    YES
                           230   1     100              NO    NO                                                    EN

TERMINAL   STIP/ TC        CSET    TSPEED CA RIC CO    BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
           PAD
           N2741 713               300       NO

DEVICE    NAME       DT/   SDT/  ABL/  DBZ/   UBL/     HN/  AUTOCON/ BR/   AB/  B1/  CI/ CT/ DLC/   EP/  LI/  PG/  PL/ SE/
                     TA    XBZ/  DBL/  UBZ/   STREAM/  DO/     HD    PRI/  BS/  B2/  CN/ DLX/ DLTO/  IN/  OP/  PA/  PW/ STAT/
                           MCI   MLI   RTS    XLY      MC            FA    ELO  ELX  ELR EBO  EBX    EBR  CP   IC   OC  LK

          SVLTTY2    CON         1     230    7        2    YES
                           230   1     100              NO    NO                                                    EN
```

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 5 of 9)

```
TIPTYPES USED FOR THIS NPU

    ASYNC

NPU         NAME      NODE      VARIANT   OPGO      DMP
            NPDAEG     6        SELF      YES       NO

   SUPLINK  LLNAME
            LINK5

  TRUNK     NAME      N1        N2     P1 P2   NOLOAD1  NOLOAD2  STATUS    FRAME
            TRNK3     NPC2550  NPDAEG   2  1     YES      YES      DI       1050

   LINE     NAME      PORT/ LTYPE  AUTO/ TIPTYPE/ DI   LSPEED/  DFL/  FRAME/ RTIME/ RCOUNT/ NSVC/
                      LCN          IMDISC  RC          XAUTO    PSN   NPVC   AL     ARSPEED  DTEA

            NPDLN1     3    A1     YES   ASYNC    NO
                       0          NO                   NO                   0    YES

   TERMINAL STIP/ TC      CSET     TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
            PAD
            N2741                              NO

   DEVICE   NAME      DT/  SDT/  ABL/  DBZ/  UBL/    HN/  AUTOCON/ BR/   AB/  B1/  CI/ CT/ DLC/   EP/ LI/ PG/ PL/ SE/
                      TA   XBZ/  DBL/  UBZ/  STREAM/ DO/       HD  PRI/  BS/  B2/  CN/ DLX/ DLTO/ IN/ OP/ PA/ PW/ STAT/
                           MCI   MLI   RTS   XLY     MC            FA    ELO  ELX  ELR EBO  EBX   EBR CP  IC  OC  LK

            STATTY1   CON        1     230    7     2    YES
                           230   1     100                    NO     NO                                                 EN

   LINE     NAME      PORT/ LTYPE  AUTO/ TIPTYPE/ DI   LSPEED/  DFL/  FRAME/ RTIME/ RCOUNT/ NSVC/
                      LCN          IMDISC  RC          XAUTO    PSN   NPVC   AL     ARSPEED  DTEA

            NPDLN2     2    A1     YES   ASYNC    NO
                       0          NO                   NO                   0    YES

   TERMINAL STIP/ TC      CSET     TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
            PAD
            N2741 M33              150         NO
```

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 6 of 9)

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	STATTY2	CON	230	1 1	230 100	7	2	YES NO	YES										EN

LINE	NAME	PORT/ LCN	LTYPE	AUTO/ IMDISC	TIPTYPE/ RC	DI	LSPEED/ XAUTO	DFL/ PSN	FRAME/ NPVC	RTIME/ AL	RCOUNT/ ARSPEED	NSVC/ DTEA
	NPDLN3	4 0	A2	YES NO	ASYNC	NO	NO			0	YES	

TERMINAL	STIP/ PAD	TC	CSET	TSPEED	CA	RIC	CO	BCF	MREC	W	CTYP	NCIR	NEN	EOF	COLLECT
	N2741					NO									

DEVICE	NAME	DT/ TA	SDT/ XBZ/ MCI	ABL/ DBL/ MLI	DBZ/ UBZ/ RTS	UBL/ STREAM/ XLY	HN/ DO/ MC	AUTOCON/ HD	BR/ PRI/ FA	AB/ BS/ ELO	B1/ B2/ ELX	CI/ CN/ ELR	CT/ DLX/ EBO	DLC/ DLTO/ EBX	EP/ IN/ EBR	LI/ OP/ CP	PG/ PA/ IC	PL/ PW/ OC	SE/ STAT/ LK
	STATTY3	CON	230	1 1	230 100	7	2	YES NO	NO										EN

TIPTYPES USED FOR THIS NPU

ASYNC

NPU	NAME	NODE	VARIANT	OPGO	DMP
	NPE2550	7	N7F	NO	YES

SUPLINK	LLNAME
	LINK6
	LINK8

COUPLER	NAME	NODE	HNAME	LOC
	CPLR3	8	HOST3	PRIMARY

LOGLINK	NAME	NCNAME	STATUS
	LINK6	NPE2550	EN
	LINK7	CPLR4	EN

COUPLER	NAME	NODE	HNAME	LOC
	CPLR4	9	SYS173	SECOND

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 7 of 9)

```
LOGLINK  NAME      NCNAME    STATUS
         LINK8     NPE2550   EN

LINE     NAME      PORT/ LTYPE  AUTO/ TIPTYPE/ DI  LSPEED/  DFL/  FRAME/ RTIME/ RCOUNT/ NSVC/
                   LCN          IMDISC RC           XAUTO    PSN   NPVC   AL    ARSPEED  DTEA

         NPELN1     2    A1     YES    ASYNC   NO
                    0           NO                   NO                      0    YES

TERMINAL STIP/  TC     CSET    TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
         PAD
         N2741  M40            1200        NO

DEVICE   NAME      DT/  SDT/  ABL/  DBZ/  UBL/     HN/   AUTOCON/ BR/  AB/  B1/  CI/ CT/  DLC/    EP/  LI/  PG/  PL/  SE/
                   TA   XBZ/  DBL/  UBZ/  STREAM/  DO/      HD    PRI/ BS/  B2/  CN/ DLX/ DLTO/   IN/  OP/  PA/  PW/  STAT/
                        MCI   MLI   RTS   XLY      MC             FA   ELO  ELX  ELR EBO  EBX     EBR  CP   IC   OC   LK

         TTY1      CON         1     230   .7       8      YES
                        230    1     100                   YES     NO                                                  EN

LINE     NAME      PORT/ LTYPE  AUTO/ TIPTYPE/ DI  LSPEED/  DFL/  FRAME/ RTIME/ RCOUNT/ NSVC/
                   LCN          IMDISC RC           XAUTO    PSN   NPVC   AL    ARSPEED  DTEA

         NPELN2     4    H1     NO     X25     NO              4    1     2    15       1
                    0           NO                   NO  CDSN      0     0    NO

TERMINAL STIP/  TC     CSET    TSPEED  CA  RIC  CO   BCF  MREC  W  CTYP  NCIR  NEN  EOF  COLLECT
         PAD
         PAD    713    ASCII               NO                   2  SVC   1     1

DEVICE   NAME      DT/  SDT/  ABL/  DBZ/  UBL/     HN/   AUTOCON/ BR/  AB/  B1/  CI/ CT/  DLC/    EP/  LI/  PG/  PL/  SE/
                   TA   XBZ/  DBL/  UBZ/  STREAM/  DO/      HD    PRI/ BS/  B2/  CN/ DLX/ DLTO/   IN/  OP/  PA/  PW/  STAT/
                        MCI   MLI   RTS   XLY      MC             FA   ELO  ELX  ELR EBO  EBX     EBR  CP   IC   OC   LK

         M713A     CON         2     225   7        NO
                        2             100                  YES     NO                                                  EN
```

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 8 of 9)

```
TIPTYPES USED FOR THIS NPU

   ASYNC     X25

NPU          NAME       NODE      VARIANT   OPGO      DMP
             NPF2550    10        N6F       NO        YES

  SUPLINK    LLNAME
             LINK9

  COUPLER    NAME       NODE      HNAME     LOC
             CPLR5      11        HOST2     PRIMARY

  LOGLINK    NAME       NCNAME    STATUS
             LINK9      NPF2550   EN

  LINE       NAME       PORT/ LTYPE  AUTO/ TIPTYPE/ DI   LSPEED/   DFL/   FRAME/ RTIME/ RCOUNT/ NSVC/
                        LCN          IMDISC RC            XAUTO     PSN    NPVC   AL     ARSPEED DTEA

             NPFLN1     3     H1     NO    X25      NO                4      1      2      15      1
                        0            NO                    NO CDSN      0      0  NO

  TERMINAL   STIP/  TC     CSET     TSPEED  CA  RIC  CO  BCF MREC W  CTYP NCIR NEN EOF COLLECT
             PAD
             PAD    M40    ASCII                NO                    2  SVC   1    1

  DEVICE     NAME   DT/  SDT/  ABL/  DBZ/  UBL/     HN/  AUTOCON/ BR/  AB/  B1/  CI/ CT/ DLC/   EP/ LI/ PG/ PL/ SE/
                    TA   XBZ/  DBL/  UBZ/  STREAM/  DO/       HD  PRI/ BS/  B2/  CN/ DLX/ DLTO/ IN/ OP/ PA/ PW/ STAT/
                         MCI   MLI   RTS   XLY      MC            FA   ELO  ELX  ELR EBO  EBX    EBR CP  IC  OC  LK

             TTY20  CON        2     225   7        11   YES
                               2     100                NO       NO                                               EN


 TIPTYPES USED FOR THIS NPU

    X25

MAXIMUM NODE NUMBER USED

   11

UNUSED NODE NUMBERS

ALL NODE NUMBERS LESS THAN  11 HAVE BEEN USED
```

Figure 11-5. File Summary Listing, Network Division for File PUBSNCF (Sheet 9 of 9)

```
      LINE  ERR  DEFINE
         1            PUBSLCF:  LFILE.
         2
         3               TITLE,PUBSLCF (NDL REF MAN SAMPLE PROGRAM).
         4
         5            ***********************************************************************
         6            *  HERE ARE ALL OF THE APPLICATION PROGRAMS INSTALLED AT THIS SITE.
         7            *  EACH HAS AN APPL STATEMENT JUST FOR DOCUMENTATION PURPOSES; ONLY
         8            *  THE STATEMENTS FOR RBF, TAF, AND RMV2 ARE NECESSARY.
         9            *  THE PRIVILEGED RESTRICTION IS NOT REQUIRED FOR THE OTHERS.
        10
        11            RBF:    APPL, PRIV=YES, UID=YES, DI=NO.
        12            IAF:    APPL, PRIV=YES, UID=NO,  DI=NO.
        13            TAF:    APPL, PRIV=YES, UID=YES, DI=NO.
        14            MCS:    APPL, PRIV=YES, UID=NO,  DI=NO.
        15            TVF:    APPL, PRIV=NO,  UID=NO,  DI=NO.
        16
        17            COMMENT   THE FOLLOWING IS A SITE-DEFINED NETWORK APPLICATION PROGRAM
        18            *         AND IS NOT IN THE SYSTEM LIBRARY:
        19
        20            RMV2:   APPL, DI=YES.     * RMV2 IS INITIALLY DISABLED.
        21
        22            ***********************************************************************
        23            *  HERE ARE ALL OF THE DEVICES REQUIRING AN AUTOMATIC OR MANDATORY
        24            *  LOGIN CONFIGURATION.
        25            *  ALL USER NAMES ARE KNOWN TO BE VALID FOR THIS HOST.
        26            *  THE DEFAULT FAMILY NAME IS SVLSYS.  SVLFAM2 IS ALSO KNOWN TO BE VALID.
        27
        28            COMMENT   THIS TERMINAL HAS DEFAULTS FOR ALL LOGIN PARAMETERS; USERS
        29            *         ARE STILL PROMPTED FOR EACH PARAMETER EXCEPT PASSWORD:
        30
        31            HASPCON:  USER, DFAM=SVLFAM2, DUSER=HASPUSR, PAPPL=RBF.
        32
        33            COMMENT   THIS TERMINAL HAS ONLY MANDATORY LOGIN PARAMETERS; USERS
        34            *         ARE NOT PROMPTED FOR ANY PARAMETERS AND ONLY RBF IS
        35            *         ACCESSIBLE FROM THIS TERMINAL:
        36
        37            BATCH3:   USER, MFAM=0, MUSER=RBFUSER, MAPPL=RBF.
        38
        39            COMMENT   THIS TERMINAL IS ALLOWED ACCESS TO SEVERAL FILE DEVICE
        40            *         FAMILIES AND THEREFORE MUST INDICATE A  FAMILY NAME.
        41            *         THE USER NAME CAN BE DEFAULTED, AND ACCESS IS ONLY
        42            *         PERMITTED TO RBF.
        43
        44            UT200A1:  USER, DFAM=NONE, DUSER=USERRBF, MAPPL=RBF.
        45
        46            COMMENT   THIS TERMINAL IS ALLOWED ACCESS TO SEVERAL FILE DEVICE
        47            *         FAMILIES BUT NEED NOT INDICATE A FAMILY NAME; FAMILY
        48            *         NAME AND THE USERNAME CAN BE DEFAULTED.  ACCESS IS
        49            *         PERMITTED ONLY TO RMV2.
        50
        51            UT200B4:  USER, DFAM=0, DUSER=USERRBF, MAPPL=RMV2.
        52
        53            COMMENT   THIS DEVICE IS ALLOWED ACCESS TO ANY APPLICATION PROGRAM,
        54            *         BUT CAN ONLY LOG INTO ONE WITH A FIXED FAMILY NAME AND
        55            *         USER NAME.
        56
        57            HQRTTY1:  USER, MFAM=0, MUSER=USER8.
        58
        59            COMMENT   THE USER OF THIS DEVICE MUST PROVIDE ALL LOGIN PARAMETERS
        60            *         EXCEPT THE LAST ONE, FOR WHICH A DEFAULT EXISTS.
        61
```

Figure 11-6. Normal Source Listing, Local Division (1 of 2)

```
62          HQRTTY2:  USER, DFAM=NONE, DUSER=NONE, PAPPL=IAF.
63
64          COMMENT   THIS DEVICE IS ALLOWED ACCESS TO ANY APPLICATION PROGRAM,
65          *         BUT CAN ONLY LOG INTO ONE WITH A FIXED USER NAME AND IS
66          *         PROMPTED FOR FAMILY NAME.
67
68          ARHTTY1:  USER, MFAM=NONE, MUSER=USER20, MAPPL=NONE.
69
70          COMMENT   THIS TERMINAL IS CONFIGURED FOR LOGIN DIALOG COMPATIBLE
71          *         WITH EARLIER VERSIONS OF NETWORK PRODUCTS.
72
73          ARHTTY2:  USER, DFAM=0.
74
75          COMMENT   THIS TERMINAL IS CONFIGURED FOR LOGIN DIALOG THAT PERMITS
76          *         ALL PARAMETERS TO BE ENTERED, BUT PROVIDES ONLY ONE DEFAULT:
77
78          SVLTTY1:  USER, DFAM=NONE, DUSER=USER30, PAPPL=NONE.
79
80          COMMENT   THIS TERMINAL IS CONFIGURED FOR LOGIN DIALOG THAT PERMITS
81          *         ALL PARAMETERS TO BE ENTERED, BUT PROVIDES TWO DEFAULTS:
82
83          SVLTTY2:  USER, DFAM=NONE, DUSER=USER40, PAPPL=IAF.
84
85          COMMENT   ALL OTHER TERMINALS ARE CONFIGURED BY OMISSION, SUCH THAT
86          *         THEIR USERS MUST ENTER ALL FOUR LOGIN PARAMETERS.
87
88          ************************************************************************
89          *  END OF LOCAL DIVISION, END OF LOCAL DEFINITION, AND END OF NETWORK
90          *  DEFINITION.
91          ************************************************************************
92
93          END.
```

Figure 11-6. Normal Source Listing, Local Division (2 of 2)

NO DEFINE COMMANDS ENCOUNTERED

Figure 11-7. DEFINE Statement Summary Listing, Local Division

```
          LINE   ERR  DEFINE
            1              PUBSLCF:  LFILE.
            2
            3                 TITLE,PUBSLCF (NDL REF MAN SAMPLE PROGRAM).
            4
            5              ***********************************************************************
            6              *  HERE ARE ALL OF THE APPLICATION PROGRAMS INSTALLED AT THIS SITE.
            7              *  EACH HAS AN APPL STATEMENT JUST FOR DOCUMENTATION PURPOSES; ONLY
            8              *  THE STATEMENTS FOR RBF, TAF, AND RMV2 ARE NECESSARY.
            9              *  THE PRIVILEGED RESTRICTION IS NOT REQUIRED FOR THE OTHERS.
           10
           11              RBF:    APPL, PRIV=YES, UID=YES, DI=NO.
           12              IAF:    APPL, PRIV=YES, UID=NO,  DI=NO.
           13              TAF:    APPL, PRIV=YES, UID=YES, DI=NO.
           14              MCS:    APPL, PRIV=YES, UID=NO,  DI=NO.
           15              TVF:    APPL, PRIV=NO,  UID=NO,  DI=NO.
           16
           17              COMMENT   THE FOLLOWING IS A SITE-DEFINED NETWORK APPLICATION PROGRAM
           18              *         AND IS NOT IN THE SYSTEM LIBRARY:
           19
           20              RMV2:   APPL, DI=YES.     * RMV2 IS INITIALLY DISABLED.
           21
           22              ***********************************************************************
           23              *  HERE ARE ALL OF THE DEVICES REQUIRING AN AUTOMATIC OR MANDATORY
           24              *  LOGIN CONFIGURATION.
           25              *  ALL USER NAMES ARE KNOWN TO BE VALID FOR THIS HOST.
           26              *  THE DEFAULT FAMILY NAME IS SVLSYS.  SVLFAM2 IS ALSO KNOWN TO BE VALID.
           27
           28              COMMENT   THIS TERMINAL HAS DEFAULTS FOR ALL LOGIN PARAMETERS; USERS
           29              *         ARE STILL PROMPTED FOR EACH PARAMETER EXCEPT PASSWORD:
           30
           31              HASPCON:  USER, DFAM=SVLFAM2, DUSER=HASPUSR, PAPPL=RBF.
           32
           33              COMMENT   THIS TERMINAL HAS ONLY MANDATORY LOGIN PARAMETERS; USERS
           34              *         ARE NOT PROMPTED FOR ANY PARAMETERS AND ONLY RBF IS
           35              *         ACCESSIBLE FROM THIS TERMINAL:
           36
           37              BATCH3:   USER, MFAM=0, MUSER=RBFUSER, MAPPL=RBF.
           38
           39              COMMENT   THIS TERMINAL IS ALLOWED ACCESS TO SEVERAL FILE DEVICE
           40              *         FAMILIES AND THEREFORE MUST INDICATE A  FAMILY NAME.
           41              *         THE USER NAME CAN BE DEFAULTED, AND ACCESS IS ONLY
           42              *         PERMITTED TO RBF.
           43
           44              UT200A1:  USER, DFAM=NONE, DUSER=USERRBF, MAPPL=RBF.
           45
           46              COMMENT   THIS TERMINAL IS ALLOWED ACCESS TO SEVERAL FILE DEVICE
           47              *         FAMILIES BUT NEED NOT INDICATE A FAMILY NAME; FAMILY
           48              *         NAME AND THE USERNAME CAN BE DEFAULTED.  ACCESS IS
           49              *         PERMITTED ONLY TO RMV2.
           50
           51              UT200B4:  USER, DFAM=0, DUSER=USERRBF, MAPPL=RMV2.
           52
           53              COMMENT   THIS DEVICE IS ALLOWED ACCESS TO ANY APPLICATION PROGRAM,
           54              *         BUT CAN ONLY LOG INTO ONE WITH A FIXED FAMILY NAME AND
           55              *         USER NAME.
           56
           57              HQRTTY1:  USER, MFAM=0, MUSER=USER8.
           58
           59              COMMENT   THE USER OF THIS DEVICE MUST PROVIDE ALL LOGIN PARAMETERS
           60              *         EXCEPT THE LAST ONE, FOR WHICH A DEFAULT EXISTS.
           61
           62              HQRTTY2:  USER, DFAM=NONE, DUSER=NONE, PAPPL=IAF.
           63
           64              COMMENT   THIS DEVICE IS ALLOWED ACCESS TO ANY APPLICATION PROGRAM,
           65              *         BUT CAN ONLY LOG INTO ONE WITH A FIXED USER NAME AND IS
           66              *         PROMPTED FOR FAMILY NAME.
           67
```

Figure 11-8. Expanded Source Listing, Local Division (Sheet 1 of 2)

```
68        ARHTTY1:  USER, MFAM=NONE, MUSER=USER20, MAPPL=NONE.
69
70        COMMENT   THIS TERMINAL IS CONFIGURED FOR LOGIN DIALOG COMPATIBLE
71        *         WITH EARLIER VERSIONS OF NETWORK PRODUCTS.
72
73        ARHTTY2:  USER, DFAM=0.
74
75        COMMENT   THIS TERMINAL IS CONFIGURED FOR LOGIN DIALOG THAT PERMITS
76        *         ALL PARAMETERS TO BE ENTERED, BUT PROVIDES ONLY ONE DEFAULT:
77
78        SVLTTY1:  USER, DFAM=NONE, DUSER=USER30, PAPPL=NONE.
79
80        COMMENT   THIS TERMINAL IS CONFIGURED FOR LOGIN DIALOG THAT PERMITS
81        *         ALL PARAMETERS TO BE ENTERED, BUT PROVIDES TWO DEFAULTS:
82
83        SVLTTY2:  USER, DFAM=NONE, DUSER=USER40, PAPPL=IAF.
84
85        COMMENT   ALL OTHER TERMINALS ARE CONFIGURED BY OMISSION, SUCH THAT
86        *         THEIR USERS MUST ENTER ALL FOUR LOGIN PARAMETERS.
87
88        **********************************************************************
89        *  END OF LOCAL DIVISION, END OF LOCAL DEFINITION, AND END OF NETWORK
90        *  DEFINITION.
91        **********************************************************************
92
93        END.
```

Figure 11-8. Expanded Source Listing, Local Division (Sheet 2 of 2)

```
              VALID LCF CREATED     06.58.11  85/08/07

  LCF FILE NAME          PUBSLCF

  APPL     NAME    PRIV    UID    STATUS   RS    MXCOPYS    KDSP    NETXFR    PRU
           RBF     YES     YES    EN       NO    1          NO      NO        NO
           IAF     YES     NO     EN       NO    1          NO      NO        NO
           TAF     YES     YES    EN       NO    1          NO      NO        NO
           MCS     YES     NO     EN       NO    1          NO      NO        NO
           TVF     NO      NO     EN       NO    1          NO      NO        NO
           RMV2    NO      NO     DI       NO    1          NO      NO        NO

  USER     NAME      FAMILY    F-STATUS   USER      U-STATUS   APPL    A-STATUS
           HASPCON   SVLFAM2   DEF        HASPUSR   DEF        RBF     PRI
           BATCH3    0         MAN        RBFUSER   MAN        RBF     MAN
           UT200A1                        USERRBF   DEF        RBF     MAN
           UT200B4   0         DEF        USERRBF   DEF        RMV2    MAN
           HQRTTY1   0         MAN        USER8     MAN
           HQRTTY2                                             IAF     PRI
           ARHTTY1                        USER20    MAN
           ARHTTY2   0         DEF
           SVLTTY1                        USER30    DEF
           SVLTTY2                        USER40    DEF        IAF     PRI
```

Figure 11-9. File Summary Listing, Local Division for File PUBSLCF

HOST 3

CHANNEL 8

COUPLER | COUPLER

PORT=2

M40 TTY CONSOLE — NPELN1 → NPE2550(65K) / N7F

PORT=4

NPELN1

PSN

EST ENTRIES

EQ70=NP,ON,0,1,4,0,1.
EQ71=NP,ON,0,1,6,0,2.

NAM INSTALLA-TION

MAXPS.

713 TERM

M40 TTY

NPFLN1

PORT=3

NPF2550(65K) / N6F

COUPLER

CHANNEL 4

HOST 2

CHANNEL 7

SYS173

MCS | NVF
RMV2 | IAF | RBF
TVF | TAF

OPERATING SYSTEM

COMMUNICATIONS SUPERVISOR

NETWORK SUPERVISOR

CHANNEL 4   CHANNEL 6

VALIDUz

NCTFid

LCF

NCF

CCP

USER 1

PUBSLCF

PUBSNCF

N2L
N3L   NPU LOAD FILE
N5F

COUPLER   COUPLER

ANY MODE 4C CONSOLE

217 BCD CONSOLE | 224 BCD CARD READER | 222 BCD LINE PRINTER

4800 BAUD

734 ASCII CONSOLE | 734-101 ASCII CARD READER | 734-201 ASCII LINE PRINTER

NPALN1

PORT=2

PORT=3

PORT=4

NPALN2

NPA2550(98K) / N2L

NPC2550(65K) / N3L

← PORT=3

NPCLN1

PORT=1   TRNK1

PORT=1   TRNK2

PORT=2

PORT=1   TRNK3

PORT=1

752 CONSOLE

MODEL 40 TELETYPE CONSOLE

9600 BAUD

1200 HASP CONSOLE †

1600 HASP CARD READER

2000 HASP PLOTTER

1600 HASP CARD READER

1200 HASP EBCDIC PRINTER

1600 HASP EBCDIC CARD PUNCH

1600 HASP CARD READER

1200 HASP EBCDIC PRINTER

NPALN3

—110 BAUD

35KSR TELETYPE

2741EBCD APL TERMINAL

—134 BAUD

2741 CORRESPOND-ENCE CODE TERMINAL

H2000 SERIES WITH ECHOPLEX

PORT=1

NPB2551(65K) / N5F

NPDAEG(65K) / SELF

← PORT=4

NPDLN3

PORT=3   PORT=4

NPBLN1 NPBLN2

PORT=3   PORT=2

NPDLN1 NPDLN2

134 BAUD   600 BAUD

150 BAUD

ANY ASYNCHRONOUS CONSOLE

MODEL 37 TELETYPE CONSOLE †

ANY ASYNCHRONOUS CONSOLE

†Input and output receives priority over other network data traffic.

Figure 11-10. Physical Configuration of Network PUBSNET
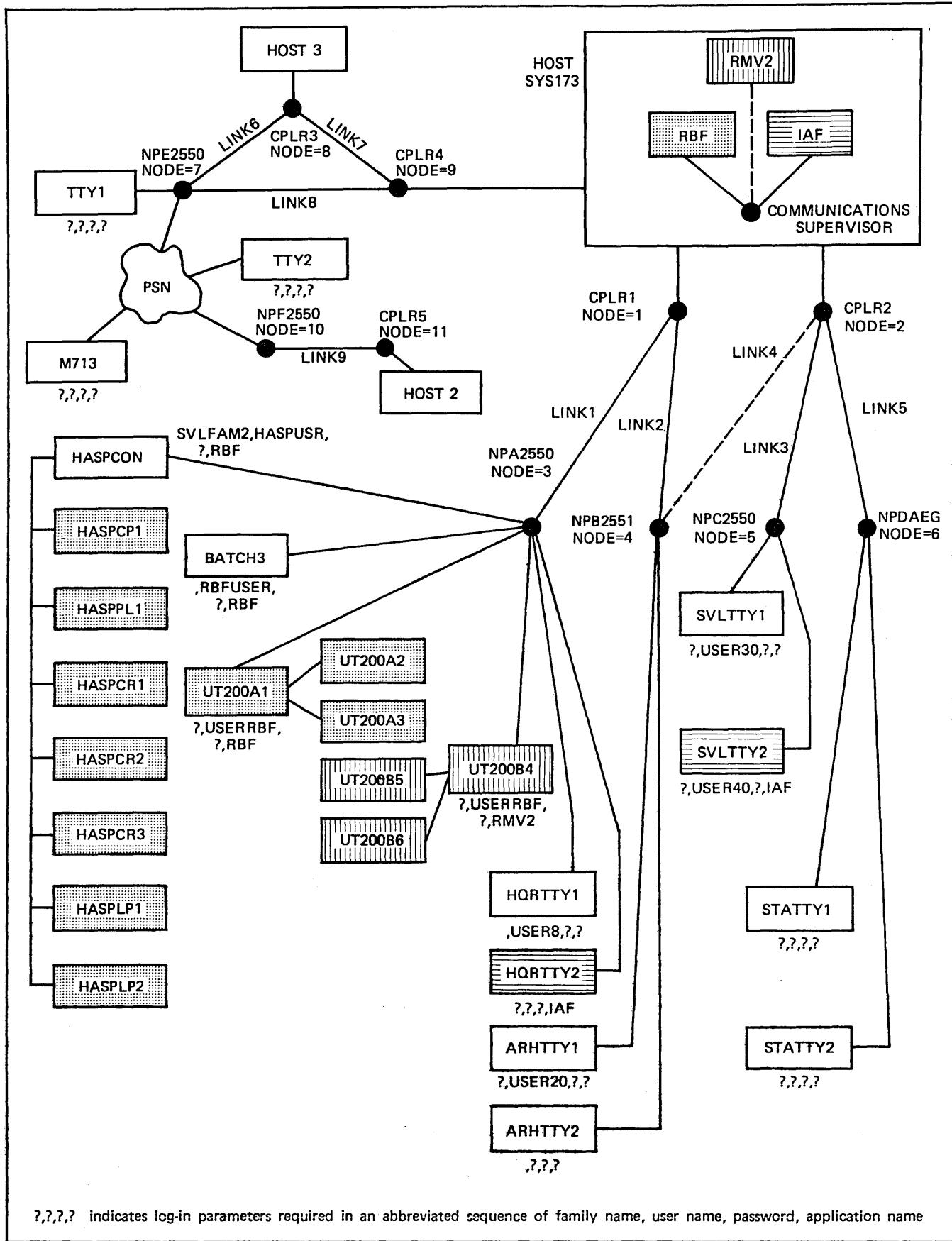
Figure 11-11. Logical Configuration of Network PUBSNET

?,?,?,? indicates log-in parameters required in an abbreviated sequence of family name, user name, password, application name

This appendix describes the code and character sets used by the operating system local batch device driver programs, magnetic tape driver programs, and network terminal communication products. This appendix does not describe how other products associate certain graphic or control characters with specific binary code values for collating or syntax processing purposes. The main text of this manual describes such associations that are relevant to the reader.

CHARACTER SETS AND CODE SETS

A character set differs from a code set. A character set is a set of graphic and/or control character symbols. A code set is a numbering system used to represent each character within a character set. Characters exist outside the computer system and communication network; codes are received, stored, retrieved, and transmitted within the computer system and network.

When this manual refers to the ASCII 128-character set or the 7-bit ASCII code set, it is referring to the character set and code set defined as the American National Standard Code for Information Interchange (ASCII, ANSI Standard X3.4-1977). References in this manual to an ASCII character set or an ASCII code set do not necessarily apply to the 128-character, 7-bit ASCII code set.

GRAPHIC AND CONTROL CHARACTERS

A graphic character can be displayed or printed. Examples of graphic characters are the characters A through Z, a blank, and the digits 0 through 9. A control character is not a graphic character; a control character initiates, modifies, or stops a control operation. An example of a control character is the backspace character, which moves the terminal carriage or cursor back one space. Although a control character is not a graphic character, some terminals use a graphic representation for control characters.

CODED AND BINARY CHARACTER DATA

Character codes can be interpreted as coded character data or as binary character data. Coded character data is converted by default from one code set representation to another as it enters or leaves the computer system; for example, data received from a terminal or sent to a magnetic tape unit is converted. Binary character data is not converted as it enters or leaves the system. Character codes are not converted when moved within the system; for example, data transferred to or from mass storage is not converted.

The distinction between coded character data and binary character data is important when reading or punching cards and when reading or writing magnetic tape. Only coded character data can be properly reproduced as characters on a line printer. Only binary character data can properly represent characters on a punched card when the data cannot be stored as display code.

The distinction between binary character data and characters represented by binary data (such as peripheral equipment instruction codes) is also important. Only binary noncharacter data can properly reproduce characters on a plotter.

CHARACTER SET TABLES

The character set tables in this appendix are designed so that the user can find the character represented by a code (such as in a dump) or find the code that represents a character. To find the character represented by a code, the user looks up the code in the column listing the appropriate code set and then finds the character on that line in the column listing the appropriate character set. To find the code that represents a character, the user looks up the character and then finds the code on the same line in the appropriate column.

NETWORK OPERATING SYSTEM

NOS supports the following character sets:

 CDC graphic 64-character set

 CDC graphic 63-character set

 ASCII graphic 64-character set

 ASCII graphic 63-character set

 ASCII graphic 95-character set

 ASCII 128-character set

Each installation must select either a 64-character set or a 63-character set. The differences between the codes of a 63-character set and the codes of a 64-character set are described under Character Set Anomalies. Any reference in this appendix to a 64-character set implies either a 63- or 64-character set unless otherwise stated.

NOS supports the following code sets to represent its character sets in central memory:

 6-bit display code

 12-bit ASCII code

 6/12-bit display code

The 6-bit display code is a set of octal codes from 00 to 77, inclusive.

The 12-bit ASCII code is the ASCII 7-bit code right-justified in a 12-bit byte. The bits are numbered from the right starting with 0; bits 0 through 6 contain the ASCII code, bits 7 through 10 contain zeros, and bit 11 distinguishes the 12-bit ASCII 0000 code from the 12-bit 0000 end-of-line byte. The octal values for the 12-bit codes are 0001 through 0177 and 4000.

The 6/12-bit display code is a combination of 6-bit codes and 12-bit codes. The octal values for the 6-bit codes are 00 through 77, excluding 74 and 76. (The interpretation of the 00 and 63 codes is described under Character Set Anomalies in this appendix.) The octal 12-bit codes begin with either 74 or 76 and are followed by a 6-bit code. Thus, 74 and 76 are escape codes and are never used as 6-bit codes within the 6/12-bit display code set. The octal values of the 12-bit codes are: 7401, 7402, 7404, 7407, and 7601 through 7677. The other 12-bit codes, 74xx and 7600, are undefined.

CHARACTER SET ANOMALIES

The operating system input/output software and some products interpret two codes differently when the installation selects a 63-character set rather than a 64-character set. If a site uses a 63-character set: the colon (:) graphic character is always represented by a 6-bit display code value of 63 octal; display code 00 is undefined (it has no associated graphic or punched card code); the percent (%) graphic does not exist, and translations produce a space (55 octal).

However, if the site uses a 64-character set, output of an octal 7404 6/12-bit display code or a 6-bit display code value of 00 produces a colon. In ASCII mode, a colon can be input only as a 7404 6/12-bit display code. Undefined 6/12-bit display codes in output files produce unpredictable results and should be avoided.

Two consecutive 6-bit display code values of 00 can be confused with the 12-bit 0000 end-of-line byte and should be avoided.

Translation of 7-bit or 12-bit ASCII to 6-bit display code causes character folding from the 128-character ASCII set to the 63- or 64-character ASCII subset, with the special character substitutions shown in figure A-1.

INTERACTIVE TERMINAL USERS

NOS supports display consoles and teletypewriters that use code sets other than 7-bit ASCII codes for communication or use graphics other than those defined in an ASCII character set. Data exchanged with such terminals is translated as described under Terminal Transmission Modes in this appendix. The following description applies only to terminals that use 7-bit ASCII codes and the ASCII character set.

ASCII Data Exchange Modes

Table A-1 shows the character sets and code sets available to an Interactive Facility (IAF) user. Table A-2 shows the octal and hexadecimal 7-bit ASCII code for each ASCII character, and can be used to convert codes from octal to hexadecimal. (Certain Terminal Interface Program commands require hexadecimal specification of a 7-bit ASCII code.)

IAF supports both normalized mode and transparent mode transmissions through the network. These transmission modes are described under Terminal Transmission Modes in this appendix. Refer to the NOS Version 2 Reference Set, Volume 3 System Commands, for additional information.

IAF treats normalized mode transmissions as coded character data; IAF converts these transmissions to or from either 6-bit or 6/12-bit display code.

IAF treats transparent mode transmissions as binary character data. Transparent mode input or output uses 12-bit bytes, with bit 11 always set to 1; for ASCII terminals, transparent mode input and output occurs in the 12-bit ASCII code shown in table A-1, but the leftmost digit is 4 instead of 0.

When the NORMAL command is in effect, IAF assumes that the ASCII graphic 64-character set is used and translates all input and output to or from display code. When the ASCII command is in effect, IAF assumes that the ASCII 128-character set is used and translates all input and output to or from 6/12-bit display code.

The IAF user can convert a 6/12-bit display code file to a 12-bit ASCII code file using the NOS FCOPY control statement. The resulting 12-bit ASCII file can be routed to a line printer but the file cannot be output through IAF.

12-Bit ASCII (Octal)	63- or 64-Character Subset			
		6-Bit Display Code (Octal)		12-Bit ASCII (Octal)
0140 (`)		74 (@)		0100 (@)
0173 ({)		6↑ ([)		0133 ([)
0174 (|)	Input	75 (\)	Output	0134 (\)
0175 (})		62 (])		0135 (])
0176 (~)		76 ( )		0136 ( )

Figure A-1. ASCII Character Folding

Terminal Transmission Modes

Coded character data can be exchanged with a conversational terminal in two transmission modes. These two modes, normalized mode and transparent mode, correspond to the types of character code editing and translation performed by the network software during input and output operations.

The terminal operator can change the input transmission mode using a terminal definition command (sometimes called a Terminal Interface Program command). The application program providing the terminal facility service can change the input or output transmission mode.

Normalized Mode Transmissions

Normalized mode is the initial and default mode used for both input and output transmissions. The network software translates normalized mode data to or from the transmission code used by the terminal into or from the 7-bit ASCII code shown in table A-2. (Tables A-1 and A-3 through A-7 are provided for use while coding an application program to run under the operating system; they do not describe character transmissions through the network.) Translation of a specific terminal transmission code to or from a specific 7-bit ASCII code depends on the terminal class in which the network software places the terminal.

The following paragraphs summarize the general case for normalized mode data code translations. This generalized description uses table A-2.

The reader can extend this generalized description by using the other tables to determine character set mapping for functions initiated from a terminal. For example, the description under Terminal Output Character Sets can be used to predict whether a lowercase ASCII character stored in 6/12-bit display code can appear on an EBCDIC or external BCD terminal; if an ASCII character passes through the network represented in 7-bit ASCII as character mode data, it probably can be represented on an EBCDIC terminal, but it is always transformed to an uppercase character on a mode 4A ASCII terminal.

Table A-2 contains the ASCII 128-character set supported by the network software. The ASCII 96-character subset in the rightmost six columns minus the deletion character (DEL) comprises the graphic 95-character subset; the DEL is not a graphic character, although some terminals graphically represent it. The graphic 64-character subset comprises the middle four columns. Only the characters in this 64-character subset have 6-bit display code equivalents.

Terminals that support an ASCII graphic 64-character subset actually use a subset of up to 96 characters, consisting of the graphic 64-character subset and the control characters of columns 1 and 2; often, the DEL character in column 7 is included. Terminals that support an ASCII graphic 95-character or 96-character subset actually might use all 128 characters.

The hexadecimal value of the 7-bit code for each character in table A-2 consists of the character's column number in the table, followed by its row number. For example, N is in row E of column 4, so

its hexadecimal value is 4E. The octal value for the code when it is right-justified in an 8-bit byte appears beneath the character graphic or mnemonic. The binary value of the code consists of the bit values shown, placed in the order given by the subscripts for the letter b; for example, N is 1001110.

Tables A-8 through A-19 show the normalized mode translations performed for each terminal class. The parity shown in the terminal transmission codes is the parity used as a default for the terminal class. The parity setting actually used by a terminal can be identified to the network software through a TIP command.

Tables A-8 through A-19 contain the graphic and control characters associated with the transmission codes used by the terminal because of the terminal class and code set in use. The network ASCII graphic and control characters shown are those of the standard ASCII character set associated with the ASCII transmission codes of table A-2.

Terminal Output Character Subsets -- Although the network supports the ASCII 128-character set, some terminals restrict output to a smaller character set. This restriction is supported by replacing the control characters in columns 0 and 1 of table A-2 with blanks to produce the ASCII graphic 95-character subset, and replacing the characters in columns 6 and 7 with the corresponding characters from columns 4 and 5, respectively, to produce the ASCII graphic 64-character subset.

Terminal Input Character Subsets and Supersets -- Although the network supports the ASCII 128-character set, some terminals restrict input to a smaller character set or permit input of a larger character set. A character input from a device using a character set other than ASCII is converted to an equivalent ASCII character; terminal characters without ASCII character equivalents are represented by the ASCII code for a space.

Site-written terminal-servicing facility programs can also cause input or output character replacement, conversion, or deletion by exchanging data with the network in 6-bit display code.

Input Restrictions -- The network software automatically deletes codes associated with terminal communication protocols or terminal hardware functions. These codes usually represent the cancel, backspace, linefeed, carriage return, and deletion characters. If paper tape support is requested, the device control 3 code also is deleted. Some of these code deletions can be suppressed by using the full-ASCII and special editing options (refer to the FA and SE terminal definition parameters in the NOS Version 2 Reference Set, Volume 3, System Commands).

Output Restrictions -- All codes sent by an application program are transmitted to the terminal. However, the 12-bit ASCII code 0037 (octal), the 6/12-bit display code 7677 (octal), and the 7-bit ASCII code 1F (hexadecimal) should be avoided in character mode output. The network software interprets the unit separator character represented by these codes as an end-of-line indicator. The processing of application program-supplied unit separators causes incorrect formatting of output and can cause loss of other output characters.

Input Parity Processing -- The network software does not preserve the parity of the terminal transmission code in the corresponding ASCII code. An ASCII code received by the terminal-servicing facility program always contains zero as its eighth bit.

Output Parity Processing -- The network software provides the parity bit setting appropriate for the terminal being serviced, even when the software is translating from ASCII character codes with zero parity bit settings.

Transparent Mode Transmissions

Transparent mode is selected separately for input and output transmissions.

During transparent mode input, the parity bit is stripped from each terminal transmission code (unless the N or I parity option has been selected by a terminal definition command), and the transmission code is placed in an 8-bit byte without translation to 7-bit ASCII code. Line transmission protocol characters are deleted from mode 4 terminal input. When the 8-bit bytes arrive in the host computer, a terminal servicing facility program can right-justify the bytes within a 12-bit byte.

During transparent mode output, processing similar to that performed for input occurs. When the host computer transmits 12-bit bytes, the leftmost 4 bits (bits 11 through 8) are discarded. The code in each 8-bit byte received by the network software is not translated. The parity bit appropriate for the terminal class is altered as indicated by the parity option in effect for the terminal. The codes are then transmitted to the terminal in bytes of a length appropriate for the terminal class. Line transmission protocol characters are inserted into mode 4 terminal output.

LOCAL BATCH USERS

Table A-3 lists the CDC graphic 64-character set, the ASCII graphic 64-character set, and the ASCII graphic 95-character set available on local batch devices. This table also lists the code sets and card keypunch codes (026 and 029) that represent the characters.

The 64-character sets use 6-bit display code as their code set; the 95-character set uses 12-bit ASCII code. The 95-character set is composed of all the characters in the ASCII 128-character set that can be printed at a line printer (refer to Line Printer Output). Only 12-bit ASCII code files can be printed using the graphic ASCII 95-character set. The 95-character set is represented by the octal 12-bit ASCII codes 0040 through 0176. An octal 12-bit ASCII code outside of the range 0040 through 0176 represents an unprintable character.

To print a 6/12-bit display code file, the user must convert the file to 12-bit ASCII code. The NOS FCOPY control statement is used for this conversion.

Line Printer Output

The printer train used on the line printer to which a file is sent determines which batch character set is printed. The following CDC print trains match the batch character sets in table A-3:

Character Set	Print Train
CDC graphic 64-character set	596-1
ASCII graphic 64-character set	596-5
ASCII graphic 95-character set	596-6

The characters of the default 596-1 print train are listed in the table A-3 column labeled CDC Graphic (64-Character Set); the 596-5 print train characters are listed in the table A-3 column labeled ASCII Graphic (64-Character Set); and the 596-6 print train characters are listed in the table A-3 column labeled ASCII Graphic (95-Character Set).

If an unprintable character exists in a line, NOS marks the condition by printing the number sign (#) in the first printable column of the line. A space replaces the unprintable character within the line.

When a transmission error occurs during the printing of a line, NOS makes up to five attempts to reprint the line. The CDC graphic print train prints a concatenation symbol (⌐→) in the first column of the repeated line following a line containing errors. The ASCII print trains print an underline (_) instead of the concatenation symbol.

After the fifth attempt, the setting of sense switch one for the batch input and output control point determines further processing. NOS either rewinds the file and returns it to the print queue, or ignores the transmission errors.

Punched Card Input and Output

A character represented by multiple punches in a single column has its punch pattern identified by numbers and hyphens. For example, the punches representing an exclamation point are identified as 11-0; this notation means both rows 11 and 0 are punched in the same column.

A multiple punch pattern that represents something other than a character is identified by numbers and slashes. For example, the punches representing the end of an input file are identified as 6/7/8/9; this notation means rows 6 through 9 are punched in the same column.

Coded character data is exchanged with card readers or card punches according to the translations shown in table A-3. As indicated in the table, other card keypunch codes are available for input of the ASCII and CDC characters [and]. NOS cannot read or punch the 95-character set as coded character data.

Each site chooses either 026 or 029 as its default keypunch code. NOS begins reading an input deck in the default code (regardless of the character set

in use). The user can specify the alternate key-
punch code by punching a 26 or 29 in columns 79 and
80 of any job card, 6/7/9 card, or 7/8/9 card. The
specified translation continues throughout the job
unless the alternate keypunch code translation is
specified on a subsequent 6/7/9 or 7/8/9 card.

A 5/7/9 card with a punch in column 1 changes
keypunch code translation if the card is read
immediately before or after a 7/8/9 card. A space
(no punch) in column 2 indicates 026 translation
mode; a 9 punch in column 2 indicates 029 transla-
tion mode. The specified translation remains in
effect until a similar 5/7/9 card or a 7/8/9 card
is encountered, or the job ends.

The 5/7/9 card also allows literal input when
4/5/6/7/8/9 is punched in column 2. Literal input
can be used to read 80-column binary character data
within a punched card deck of coded character data.

Literal cards are stored with each column repre-
sented in a 12-bit byte (a row 12 punch is repre-
sented by a 1 in bit 11, row 11 by a 1 in bit 10,
row 0 by a 1 in bit 9, and rows 1 through 9 by 1's
in bits 8 through 0 of the byte), using 16 central
memory words per card. Literal input cards are
read until another 5/7/9 card with 4/5/6/7/8/9
punched in column 2 is read. The next card can
specify a new conversion mode.

If the card following the 5/7/9, 6/7/9, or 7/8/9
card has a 7 and a 9 punched in column 1, the sec-
tion of the job deck following it contains system
binary cards (as described in the NOS Version 2
Reference Set, Volume 3, System Commands).

REMOTE BATCH USERS

Remote batch console input and output is restricted
to character mode transmission. Character mode is
described under Terminal Transmission Modes in this
appendix.

The abilities to select alternate keypunch code
translations, to read binary cards, to output
plotter files, and to print lowercase characters
depend upon the remote terminal equipment. Remote
batch terminal support under NOS is described in
the Remote Batch Facility (RBF) reference manual.

MAGNETIC TAPE USERS

The character and code sets used for reading and
writing magnetic tapes depend on whether coded or
binary data is read or written and on whether the
tape is 7-track or 9-track.

Coded Data Exchanges

Coded character data to be copied from mass storage
to magnetic tape is assumed to be stored in a
63- or 64-character 6-bit display code. The oper-
ating system magnetic tape driver program converts
the data to 6-bit external BCD code when writing a
coded 7-track tape and to 7-bit ASCII or 8-bit
EBCDIC code (as specified on the tape assignment
statement) when writing a coded 9-track tape.

Coded character data copied to mass storage from
magnetic tape is stored in a 63- or 64-character
6-bit display code. The operating system magnetic
tape driver program converts the data from 6-bit
external BCD code when reading a coded 7-track tape
and from 7-bit ASCII or 8-bit EBCDIC code (as
specified on the tape assignment statement) when
reading a coded 9-track tape.

To read and write lowercase character 7-bit ASCII
or 8-bit EBCDIC codes or to read and write control
codes, the user must assign a 7-track or 9-track
tape in binary mode.

Seven-Track Tape Input and Output

Table A-4 shows the code and character set conver-
sions between 6-bit external BCD and 6-bit display
code for 7-track tapes. Because only 63 characters
can be represented in 7-track even parity, one of
the 64 display codes is lost in conversion to and
from external BCD code.

Figure A-2 shows the differences in 7-track tape
conversion that depend on whether the system uses
the 63-character or 64-character set. The ASCII
character for the specified character code is shown
in parentheses. The output arrows show how the
6-bit display code changes when it is written on
tape in external BCD. The input arrows show how
the external BCD code changes when the tape is read
and converted to display code.

63-Character Set				
Display Code		External BCD		Display Code
00		16 (%)		00
33 (0)	Output	12 (0)	Input	33 (0)
63 (:)	⟶	12 (0)	⟶	33 (0)
64-Character Set				
Display Code		External BCD		Display Code
00 (:)		12 (0)		33 (0)
33 (0)	Output	12 (0)	Input	33 (0)
63 (%)	⟶	16 (%)	⟶	63 (%)

Figure A-2. Magnetic Tape Code Conversions

Nine-Track Tape Input and Output

Table A-5 lists the conversions between the 7-bit
ASCII code used on the tape and the 6-bit display
code used within the system. Table A-6 lists the
conversions between the 8-bit EBCDIC code used on
the tape and the 6-bit display code used within the
system.

When an ASCII or EBCDIC code representing a lower-
case character is read from a 9-track magnetic
tape, it is converted to its uppercase character

6-bit display code equivalent. Any EBCDIC code not listed in table A-6 is converted to display code 55 (octal) and becomes a space. Any code between 80 (hexadecimal) and FF (hexadecimal) read from an ASCII tape is converted to display code 00.

Binary Character Data Exchanges

Binary character data exchanged between central memory files and magnetic tape is transferred as a string of bytes without conversion of the byte contents. The grouping of the bytes and the number of bits in each byte depend on whether 7-track or 9-track tape is being used.

Seven-Track Tape Input and Output

Each binary data character code written to or read from 7-track magnetic tape is assumed to be stored in a 6-bit byte, such as the system uses for 63- or 64-character 6-bit display code. Seven-bit ASCII and 8-bit EBCDIC codes can only be read from or written to 7-track magnetic tape as binary character data if each code is stored within a 12-bit byte as if it were two character codes.

Nine-Track Tape Input and Output

Each binary data character code exchanged between central memory files and 9-track magnetic tape is assumed to be stored in an 8-bit or 12-bit byte.

During such binary data transfers, the 6/12-bit display codes and 12-bit ASCII codes shown in table A-1, the 7-bit ASCII codes shown in table A-2, or or the 8-bit hexadecimal EBCDIC codes shown in table A-7 can be read or written. The 7-bit ASCII codes and 8-bit EBCDIC codes can be exchanged either in an unformatted form or right-justified within a zero-filled 12-bit byte of memory.

When 9-track tape is written, every pair of 12-bit memory bytes becomes three 8-bit tape bytes; when 9-track tape is read, every three 8-bit tape bytes become a pair of 12-bit memory bytes. Because of the 12-bit byte pairs, codes not packed into 12-bit bytes are exchanged in their unpacked form, while codes packed in 12-bit bytes are exchanged in packed form.

When an odd number of central memory words is read or written, the lower four bits of the last 8-bit byte (bits 0 through 3 of the last word) are not used. For example, three central memory words are written on tape as 22 8-bit bytes (7.5 pairs of 12-bit bytes) and the remaining four bits are ignored.

CODE CONVERSION AIDS

Table A-7 contains the octal values of each 8-bit EBCDIC code right-justified in a 12-bit byte with zero fill. This 12-bit EBCDIC code can be produced or read using the FORM and 8-Bit Subroutines utilities.

Character Sets		Code Sets		
ASCII Graphic (64-Character Set)	ASCII Character (128-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code†	Octal 12-Bit ASCII Code
: colon††		00††		
A	A	01	01	0101
B	B	02	02	0102
C	C	03	03	0103
D	D	04	04	0104
E	E	05	05	0105
F	F	06	06	0106
G	G	07	07	0107
H	H	10	10	0110
I	I	11	11	0111
J	J	12	12	0112
K	K	13	13	0113
L	L	14	14	0114
M	M	15	15	0115
N	N	16	16	0116
O	O	17	17	0117
P	P	20	20	0120
Q	Q	21	21	0121
R	R	22	22	0122
S	S	23	23	0123
T	T	24	24	0124
U	U	25	25	0125
V	V	26	26	0126
W	W	27	27	0127
X	X	30	30	0130
Y	Y	31	31	0131
Z	Z	32	32	0132
0	0	33	33	0060
1	1	34	34	0061
2	2	35	35	0062
3	3	36	36	0063
4	4	37	37	0064
5	5	40	40	0065
6	6	41	41	0066
7	7	42	42	0067
8	8	43	43	0070
9	9	44	44	0071
+ plus	+ plus	45	45	0053
- hyphen (minus)	- hyphen (minus)	46	46	0055
* asterisk	* asterisk	47	47	0052
/ slant	/ slant	50	50	0057
(opening parenthesis	(opening parenthesis	51	51	0050
) closing parenthesis	) closing parenthesis	52	52	0051
$ dollar sign	$ dollar sign	53	53	0044
= equals	= equals	54	54	0075
space	space	55	55	0040
, comma	, comma	56	56	0054
. period	. period	57	57	0056
# number sign	# number sign	60	60	0043
[opening bracket	[opening bracket	61	61	0133
] closing bracket	] closing bracket	62	62	0135
% percent sign††	% percent sign††	63††	63††	0045
" quotation mark	" quotation mark	64	64	0042
_ underline	_ underline	65	65	0137
! exclamation point	! exclamation point	66	66	0041
& ampersand	& ampersand	67	67	0046
' apostrophe	' apostrophe	70	70	0047
? question mark	? question mark	71	71	0077

Character Sets		Code Sets		
ASCII Graphic (64-Character Set)	ASCII Character (128-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code†	Octal 12-Bit ASCII Code
< less than	< less than	72	72	0074
> greater than	> greater than	73	73	0076
@ commmercial at	@ commercial at	74††	7401††	0100
\ reverse slant	\ reverse slant	75	75	0134
∧ circumflex		76		
; semicolon	; semicolon	77	77	0073
	∧ circumflex	76††	7402	0136
	: colon††	74††	7404††	0072
	` grave accent		7407	0140
	a		7601	0141
	b		7602	0142
	c		7603	0143
	d		7604	0144
	e		7605	0145
	f		7606	0146
	g		7607	0147
	h		7610	0150
	i		7611	0151
	j		7612	0152
	k		7613	0153
	l		7614	0154
	m		7615	0155
	n		7616	0156
	o		7617	0157
	p		7620	0160
	q		7621	0161
	r		7622	0162
	s		7623	0163
	t		7624	0164
	u		7625	0165
	v		7626	0166
	w		7627	0167
	x		7630	0170
	y		7631	0171
	z		7632	0172
	{ opening brace	61††	7633	0173
	| vertical line	75††	7634	0174
	} closing brace	62††	7635	0175
	~ tilde	76††	7636	0176
	NUL		7640	4000
	SOH		7641	0001
	STX		7642	0002
	ETX		7643	0003
	EOT		7644	0004
	ENQ		7645	0005
	ACK		7646	0006
	BEL		7647	0007
	BS		7650	0010
	HT		7651	0011
	LF		7652	0012
	VT		7653	0013
	FF		7654	0014
	CR		7655	0015
	SO		7656	0016
	SI		7657	0017
	DEL		7637	0177
	DLE		7660	0020

Character Sets		Code Sets		
ASCII Graphic (64-Character Set)	ASCII Character (128-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code†	Octal 12-Bit ASCII Code
	DC1		7661	0021
	DC2		7662	0022
	DC3		7663	0023
	DC4		7664	0024
	NAK		7665	0025
	SYN		7666	0026
	ETB		7667	0027
	CAN		7670	0030
	EM		7671	0031
	SUB		7672	0032
	ESC		7673	0033
	FS		7674	0034
	GS		7675	0035
	RS		7676	0036
	US		7677	0037

†Available only on NOS.

††Character or code interpretation depends on context. Refer to Character Set Anomalies in the text.

TABLE A-2. 7-BIT ASCII CODE AND CHARACTER SETS

◄─── 128-Character Set ───►							
		◄─── 96-Character Subset ───►					
		◄─ Graphic 64-Character Subset ─►					
b7 → 0	0	0	0	1	1	1	1
b6 → 0	0	1	1	0	0	1	1
b5 → 0	1	0	1	0	1	0	1

Bits: b4 b3 b2 b1 — Row / Column →

b4 b3 b2 b1	Row	0	1	2	3	4	5	6	7
0 0 0 0	0	NUL 000	DLE 020	SP 040	0 060	@ 100	P 120	` 140	p 160
0 0 0 1	1	SOH 001	DC1 021	! 041	1 061	A 101	Q 121	a 141	q 161
0 0 1 0	2	STX 002	DC2 022	" 042	2 062	B 102	R 122	b 142	r 162
0 0 1 1	3	ETX 003	DC3 023	# 043	3 063	C 103	S 123	c 143	s 163
0 1 0 0	4	EOT 004	DC4 024	$ 044	4 064	D 104	T 124	d 144	t 164
0 1 0 1	5	ENQ 005	NAK 025	% 045	5 065	E 105	U 125	e 145	u 165
0 1 1 0	6	ACK 006	SYN 026	& 046	6 066	F 106	V 126	f 146	v 166
0 1 1 1	7	BEL 007	ETB 027	' 047	7 067	G 107	W 127	g 147	w 167
1 0 0 0	8	BS 010	CAN 030	(050	8 070	H 110	X 130	h 150	x 170
1 0 0 1	9	HT 011	EM 031	) 051	9 071	I 111	Y 131	i 151	y 171
1 0 1 0	A	LF 012	SUB 032	* 052	: 072	J 112	Z 132	j 152	z 172
1 0 1 1	B	VT 013	ESC 033	+ 053	; 073	K 113	[133	k 153	{ 173
1 1 0 0	C	FF 014	FS 034	, 054	< 074	L 114	\ 134	l 154	| 174
1 1 0 1	D	CR 015	GS 035	- 055	= 075	M 115	] 135	m 155	} 175
1 1 1 0	E	SO 016	RS 036	. 056	> 076	N 116	^ 136	n 156	~ 176
1 1 1 1	F	SI 017	US 037	/ 057	? 077	O 117	_ 137	o 157	DEL† 177

†The graphic 95-character subset does not include DEL; refer to Terminal Transmission Modes in the text.

LEGEND:

Numbers under characters are the octal values for the 7-bit character codes used within the network.

Character Sets			Code Sets			Card Keypunch Code	
CDC Graphic (64-Character Set)	ASCII Graphic (64-Character Set)	ASCII Graphic (95-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code†	Octal 12-Bit ASCII Code	026	029
: colon††	: colon††		00††			8-2	8-2
A	A	A	01	01	0101	12-1	12-1
B	B	B	02	02	0102	12-2	12-2
C	C	C	03	03	0103	12-3	12-3
D	D	D	04	04	0104	12-4	12-4
E	E	E	05	05	0105	12-5	12-5
F	F	F	06	06	0106	12-6	12-6
G	G	G	07	07	0107	12-7	12-7
H	H	H	10	10	0110	12-8	12-8
I	I	I	11	11	0111	12-9	12-9
J	J	J	12	12	0112	11-1	11-1
K	K	K	13	13	0113	11-2	11-2
L	L	L	14	14	0114	11-3	11-3
M	M	M	15	15	0115	11-4	11-4
N	N	N	16	16	0116	11-5	11-5
O	O	O	17	17	0117	11-6	11-6
P	P	P	20	20	0120	11-7	11-7
Q	Q	Q	21	21	0121	11-8	11-8
R	R	R	22	22	0122	11-9	11-9
S	S	S	23	23	0123	0-2	0-2
T	T	T	24	24	0124	0-3	0-3
U	U	U	25	25	0125	0-4	0-4
V	V	V	26	26	0126	0-5	0-5
W	W	W	27	27	0127	0-6	0-6
X	X	X	30	30	0130	0-7	0-7
Y	Y	Y	31	31	0131	0-8	0-8
Z	Z	Z	32	32	0132	0-9	0-9
0	0	0	33	33	0060	0	0
1	1	1	34	34	0061	1	1
2	2	2	35	35	0062	2	2
3	3	3	36	36	0063	3	3
4	4	4	37	37	0064	4	4
5	5	5	40	40	0065	5	5
6	6	6	41	41	0066	6	6
7	7	7	42	42	0067	7	7
8	8	8	43	43	0070	8	8
9	9	9	44	44	0071	9	9
+ plus	+ plus	+ plus	45	45	0053	12	12-8-6
- hyphen (minus)	- hyphen (minus)	- hyphen (minus)	46	46	0055	11	11
* asterisk	* asterisk	* asterisk	47	47	0052	11-8-4	11-8-4
/ slant	/ slant	/ slant	50	50	0057	0-1	0-1
(open. paren.	(open. paren.	(open. paren.	51	51	0050	0-8-4	12-8-5
) clos. paren.	) clos. paren.	) clos. paren.	52	52	0051	12-8-4	11-8-5
$ dollar sign	$ dollar sign	$ dollar sign	53	53	0044	11-8-3	11-8-3
= equals	= equals	= equals	54	54	0075	8-3	8-6
space	space	space	55	55	0040	no punch	no punch
, comma	, comma	, comma	56	56	0054	0-8-3	0-8-3
. period	. period	. period	57	57	0056	12-8-3	12-8-3
≡ equivalence	# number sign	# number sign	60	60	0043	0-8-6	8-3
[open. bracket	[open. bracket	[open. bracket	61	61	0133	8-7	12-8-2 or 12-0†††
] clos. bracket	] clos. bracket	] clos. bracket	62	62	0135	0-8-2	11-8-2 or 11-0†††
% percent sign††	% percent sign††	% percent sign††	63††	63††	0045	8-6	0-8-4

TABLE A-3. LOCAL BATCH DEVICE CHARACTER SETS (Contd)

Character Sets			Code Sets			Card Keypunch Code	
CDC Graphic (64-Character Set)	ASCII Graphic (64-Character Set)	ASCII Graphic (95-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code†	Octal 12-Bit ASCII Code	026	029
≠ not equals	" quotation mark	" quotation mark	64	64	0042	8-4	8-7
┌→ concatenation.	_ underline	_ underline	65	65	0137	0-8-5	0-8-5
∨ logical OR	⊤ exclamation pt.	⊤ exclamation pt.	66	66	0041	11-0 or 11-8-2§	12-8-7 or 11-0§
∧ logical AND	& ampersand	& ampersand	67	67	0046	0-8-7	12
↑ superscript	′ apostrophe	′ apostrophe	70	70	0047	11-8-5	8-5
↓ subscript	? question mark	? question mark	71	71	0077	11-8-6	0-8-7
< less than	< less than	< less than	72	72	0074	12-0 or 12-8-2§	12-8-4 or 12-0§
> greater than	> greater than	> greater than	73	73	0076	11-8-7	0-8-6
≤ less/equal	@ commercial at	@ commercial at	74††	7401††	0100	8-5	8-4
≥ greater/equal	\ reverse slant	\ reverse slant	75	75	0134	12-8-5	0-8-2
¬ logical NOT	∧ circumflex		76			12-8-6	11-8-7
; semicolon	; semicolon	; semicolon	77	77	0073	12-8-7	11-8-6
		∧ circumflex	76††	7402	0136		
		: colon††		7404††	0072		
		` grave accent	74††	7407	0140		
		a		7601	0141		
		b		7602	0142		
		c		7603	0143		
		d		7604	0144		
		e		7605	0145		
		f		7606	0146		
		g		7607	0147		
		h		7610	0150		
		i		7611	0151		
		j		7612	0152		
		k		7613	0153		
		l		7614	0154		
		m		7615	0155		
		n		7616	0156		
		o		7617	0157		
		p		7620	0160		
		q		7621	0161		
		r		7622	0162		
		s		7623	0163		
		t		7624	0164		
		u		7625	0165		
		v		7626	0166		
		w		7627	0167		
		x		7630	0170		
		y		7631	0171		
		z		7632	0172		
		{ open. brace	61†††	7633	0173		
			vertical line	75†††	7634	0174		
		} clos. brace	62†††	7635	0175		
		~ tilde	76†††	7636	0176		

†Available only on NOS.

††Character or code interpretation depends on context. Refer to Character Set Anomalies in the text.

†††Available for input only, on NOS.

§Available for input only, on NOS/BE or SCOPE 2.

External BCD	ASCII Character	Octal 6-Bit Display Code	External BCD	ASCII Character	Octal 6-Bit Display Code
01	1	34	40	– hyphen (minus)	46
02	2	35	41	J	12
03	3	36	42	K	13
04	4	37	43	L	14
05	5	40	44	M	15
06	6	41	45	N	16
07	7	42	46	O	17
10	8	43	47	P	20
11	9	44	50	Q	21
12†	0	33	51	R	22
13	= equals	54	52	! exclamation point	66
14	" quotation mark	64	53	$ dollar sign	53
15	@ commercial at	74	54	* asterisk	47
16†	% percent sign	63	55	' apostrophe	70
17	[opening bracket	61	56	? question mark	71
20	space	55	57	> greater than	73
21	/ slant	50	60	+ plus	45
22	S	23	61	A	01
23	T	24	62	B	02
24	U	25	63	C	03
25	V	26	64	D	04
26	W	27	65	E	05
27	X	30	66	F	06
30	Y	31	67	G	07
31	Z	32	70	H	10
32	] closing bracket	62	71	I	11
33	, comma	56	72	< less than	72
34	(opening parenthesis	51	73	. period	57
35	_ underline	65	74	) closing parenthesis	52
36	# number sign	60	75	\ reverse slant	75
37	& ampersand	67	76	^ caret	76
			77	; semicolon	77

†As the text explains, conversion of these codes depends on whether the tape is read or written.

ASCII					6-Bit Display Code[†††]	
Code Conversion[†]		Character and Code Conversion[††]				
Code (Hex)	Character	Code (Hex)	Character		ASCII Character	Code (Octal)
20	space	00	NUL		space	55
21	! exclamation point	7D	} closing brace		! exclamation point	66
22	" quotation mark	02	STX		" quotation mark	64
23	# number sign	03	ETX		# number sign	60
24	$ dollar sign	04	EOT		$ dollar sign	53
25	% percent sign[§]	05	ENQ		% percent sign[§]	63[§]
26	& ampersand	06	ACK		& ampersand	67
27	' apostrophe	07	BEL		' apostrophe	70
28	(opening parenthesis	08	BS		(opening parenthesis	51
29	) closing parenthesis	09	HT		) closing parenthesis	52
2A	* asterisk	0A	LF		* asterisk	47
2B	+ plus	0B	VT		+ plus	45
2C	, comma	0C	FF		, comma	56
2D	- hyphen (minus)	0D	CR		- hyphen (minus)	46
2E	. period	0E	SO		. period	57
2F	/ slant	0F	SI		/ slant	50
30	0	10	DLE		0	33
31	1	11	DC1		1	34
32	2	12	DC2		2	35
33	3	13	DC3		3	36
34	4	14	DC4		4	37
35	5	15	NAK		5	40
36	6	16	SYN		6	41
37	7	17	ETB		7	42
38	8	18	CAN		8	43
39	9	19	EM		9	44
3A	: colon[§]	1A	SUB		: colon[§]	00[§]
3B	; semicolon	1B	ESC		; semicolon	77
3C	< less than	7B	{ opening brace		< less than	72
3D	= equals	1D	GS		= equals	54
3E	> greater than	1E	RS		> greater than	73
3F	? question mark	1F	US		? question mark	71
40	@ commercial at	60	` grave accent		@ commercial at	74
41	A	61	a		A	01
42	B	62	b		B	02
43	C	63	c		C	03
44	D	64	d		D	04
45	E	65	e		E	05
46	F	66	f		F	06

ASCII				6-Bit Display Code[†††]	
Code Conversion[†]		Character and Code Conversion[††]			
Code (Hex)	Character	Code (Hex)	Character	ASCII Character	Code (Octal)
47	G	67	g	G	07
48	H	68	h	H	10
49	I	69	i	I	11
4A	J	6A	j	J	12
4B	K	6B	k	K	13
4C	L	6C	l	L	14
4D	M	6D	m	M	15
4E	N	6E	n	N	16
4F	O	6F	o	O	17
50	P	70	p	P	20
51	Q	71	q	Q	21
52	R	72	r	R	22
53	S	73	s	S	23
54	T	74	t	T	24
55	U	75	u	U	25
56	V	76	v	V	26
57	W	77	w	W	27
58	X	78	x	X	30
59	Y	79	y	Y	31
5A	Z	7A	z	Z	32
5B	[opening bracket	1C	FS	[opening bracket	61
5C	\ reverse slant	7C	| vertical line	\ reverse slant	75
5D	] closing bracket	01	SOH	] closing bracket	62
5E	⌃ caret	7E	~ tilde	⌃ caret	76
5F	_ underline	7F	DEL	_ underline	65

[†]When these characters are copied from or to a tape, the characters remain the same and the code changes from or to ASCII to or from display code.

[††]These characters do not exist in display code. When the characters are copied from a tape, each ASCII character is changed to an alternate display code character. The corresponding codes are also changed. Example: When the system copies a lowercase a, 61 (hexadecimal), from tape, it writes an uppercase A, 01 (octal).

[†††]A display code space always translates to an ASCII space.

[§]Character or code interpretation depends on context. Refer to Character Set Anomalies in the text.

EBCDIC				6-Bit Display Code†††	
Code Conversion†		Character and Code Conversion††			
Code (Hex)	Character	Code (Hex)	Character	ASCII Character	Code (Octal)
40	space	00	NUL	space	55
4A	¢ cent sign	1C	IFS	[opening bracket	61
4B	. period	0E	SO	. period	57
4C	< less than	C0	{ opening brace	< less than	72
4D	(opening parenthesis	16	BS	(opening parenthesis	51
4E	+ plus	0B	VT	+ plus	45
4F		vertical line	D0	} closing brace	! exclamation point	66
50	& ampersand	2E	ACK	& ampersand	67
5A	! exclamation point	01	SOH	] closing bracket	62
5B	$ dollar sign	37	EOT	$ dollar sign	53
5C	* asterisk	25	LF	* asterisk	47
5D	) closing parenthesis	05	HT	) closing parenthesis	52
5E	; semicolon	27	ESC	; semicolon	77
5F	¬ logical NOT	A1	~ tilde	^ caret	76
60	- hyphen (minus)	0D	CR	- hyphen (minus)	46
61	/ slant	0F	SI	/ slant	50
6B	, comma	0C	FF	, comma	56
6C	% percent sign§	2D	ENQ	% percent sign§	63§
6D	_ underline	07	DEL	_ underline	65
6E	> greater than	1E	IRS	> greater than	73
6F	? question mark	1F	IUS	? question mark	71
7A	: colon§	3F	SUB	: colon§	00§
7B	# number sign	03	ETX	# number sign	60
7C	@ commercial at	79	\ reverse slant	@ commercial at	74
7D	' apostrophe	2F	BEL	' apostrophe	70
7E	= equals	1D	IGS	= equals	54
7F	" quotation mark	02	STX	" quotation mark	64
C1	A	81	a	A	01
C2	B	82	b	B	02
C3	C	83	c	C	03
C4	D	84	d	D	04
C5	E	85	e	E	05
C6	F	86	f	F	06
C7	G	87	g	G	07
C8	H	88	h	H	10
C9	I	89	i	I	11
D1	J	91	j	J	12
D2	K	92	k	K	13
D3	L	93	l	L	14

EBCDIC				6-Bit Display Code†††	
Code Conversion†		Character and Code Conversion††			
Code (Hex)	Character	Code (Hex)	Character	ASCII Character	Code (Octal)
D4	M	94	m	M	15
D5	N	95	n	N	16
D6	O	96	o	O	17
D7	P	97	p	P	20
D8	Q	98	q	Q	21
D9	R	99	r	R	22
E0	\ reverse slant	6A	| vertical line	\ reverse slant	75
E2	S	A2	s	S	23
E3	T	A3	t	T	24
E4	U	A4	u	U	25
E5	V	A5	v	V	26
E6	W	A6	w	W	27
E7	X	A7	x	X	30
E8	Y	A8	y	Y	31
E9	Z	A9	z	Z	32
F0	0	10	DLE	0	33
F1	1	11	DC1	1	34
F2	2	12	DC2	2	35
F3	3	13	TM	3	36
F4	4	3C	DC4	4	37
F5	5	3D	NAK	5	40
F6	6	32	SYN	6	41
F7	7	26	ETB	7	42
F8	8	18	CAN	8	43
F9	9	19	EM	9	44

†When these characters are copied from or to a tape, the characters remain the same (except EBCDIC codes 4A (hexadecimal), 4F (hexadecimal), 5A (hexadecimal), and 5F (hexadecimal)) and the code changes from or to EBCDIC to or from display code.

††These characters do not exist in display code. When the characters are copied from a tape, each EBCDIC character is changed to an alternate display code character. The corresponding codes are also changed. Example: When the system copies a lowercase a, 81 (hexadecimal), from tape, it writes an uppercase A, 01 (octal).

†††A display code space always translates to an EBCDIC space.

§Character or code interpretation depends on context. Refer to Character Set Anomalies in the text.

Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic or Control Character[†]	Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic or Control Character[†]	Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic or Control Character[†]
00	0000	NUL	4A	0112	¢ cent sign	A7	0247	x
01	0001	SOH	4B	0113	. period	A8	0250	y
02	0002	STX	4C	0114	< less than	A9	0251	z
03	0003	ETX	4D	0115	(open. paren.	AA	0252	undefined
04	0004	PF	4E	0116	+ plus	thru	thru	
05	0005	HT	4F	0117	| logical OR	BF	0277	undefined
06	0006	LC	50	0120	& ampersand	C0	0300	{ open. brace
07	0007	DEL	51	0121	undefined	C1	0301	A
08	0010	undefined	thru	thru		C2	0302	B
09	0011	undefined	59	0131	undefined	C3	0303	C
0A	0012	SMM	5A	0132	! exclam. point	C4	0304	D
0B	0013	VT	5B	0133	$ dollar sign	C5	0305	E
0C	0014	FF	5C	0134	* asterisk	C6	0306	F
0D	0015	CR	5D	0135	) clos. paren.	C7	0307	G
0E	0016	SO	5E	0136	; semicolon	C8	0310	H
0F	0017	SI	5F	0137	¬ logical NOT	C9	0311	I
10	0020	DLE	60	0140	− minus	CA	0312	undefined
11	0021	DC1	61	0141	/ slant	CB	0313	undefined
12	0022	DC2	62	0142	undefined	CC	0314	♪
13	0023	TM	thru	thru		CD	0315	undefined
14	0024	RES	69	0151	undefined	CE	0316	¥
15	0025	NL	6A	0152	¦ vertical line	CF	0317	undefined
16	0026	BS	6B	0153	, comma	D0	0320	} clos. brace
17	0027	IL	6C	0154	% percent sign	D1	0321	J
18	0030	CAN	6D	0155	_ underline	D2	0322	K
19	0031	EM	6E	0156	> greater than	D3	0323	L
1A	0032	CC	6F	0157	? question mark	D4	0324	M
1B	0033	CU1	70	0160	undefined	D5	0325	N
1C	0034	IFS	thru	thru		D6	0326	O
1D	0035	IGS	78	0170	undefined	D7	0327	P
1E	0036	IRS	79	0171	` grave accent	D8	0330	Q
1F	0037	IUS	7A	0172	: colon	D9	0331	R
20	0040	DS	7B	0173	# number sign	DA	0332	undefined
21	0041	SOS	7C	0174	@ commercial at	thru	thru	
22	0042	FS	7D	0175	' apostrophe	DF	0337	undefined
23	0043	undefined	7E	0176	= equals	E0	0340	\ reverse slant
24	0044	BYP	7F	0177	" quotation mark	E1	0341	undefined
25	0045	LF	80	0200	undefined	E2	0342	S
26	0046	ETBB	81	0201	a	E3	0343	T
27	0047	ESCE	82	0202	b	E4	0344	U

Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic or Control Character[†]	Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic or Control Character[†]	Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic or Control Character[†]
28	0050	undefined	83	0203	c	E5	0345	V
29	0051	undefined	84	0204	d	E6	0346	W
2A	0052	SM	85	0205	e	E7	0347	X
2B	0053	CU2	86	0206	f	E8	0350	Y
2C	0054	undefined	87	0207	g	E9	0351	Z
2D	0055	ENQ	88	0210	h	EA	0352	undefined
2E	0056	ACK	89	0211	i	EB	0353	undefined
2F	0057	BEL	8A	0212	undefined	EC	0354	⊣
30	0060	undefined	thru	thru		ED	0355	undefined
31	0061	undefined	90	0220	undefined	thru	thru	
32	0062	SYN	91	0221	j	EF	0357	undefined
33	0063	undefined	92	0222	k	F0	0360	0
34	0064	PN	93	0223	l	F1	0361	1
35	0065	RS	94	0224	m	F2	0362	2
36	0066	UC	95	0225	n	F3	0363	3
37	0067	EOT	96	0226	o	F4	0364	4
38	0070	undefined	97	0227	p	F5	0365	5
39	0071	undefined	98	0230	q	F6	0366	6
3A	0072	undefined	99	0231	r	F7	0367	7
3B	0073	CU3	9A	0232	undefined	F8	0370	8
3C	0074	DC4	thru	thru		F9	0372	9
3D	0075	NAK	A0	0240	undefined	FA	0372	| vertical line
3E	0076	undefined	A1	0241	~ tilde	FB	0373	undefined
3F	0077	SUB	A2	0242	s	thru	thru	
40	0100	space	A3	0243	t	FF	0377	undefined
41	0101	undefined	A4	0244	u			
thru	thru		A5	0245	v			
49	0111	undefined	A6	0246	w			

[†]Graphic characters shown are those used on the IBM System/370 standard (PN) print train. Other devices support subsets or variations of this character graphic set.

TABLE A-8. CHARACTER CODE TRANSLATIONS, CONSOLES IN TERMINAL CLASSES 9, 14, 16, 17, AND 18
(HASP, HPRE, 2780, 3780, AND 3270)

Terminal EBCDIC				Network ASCII (Normalized Mode Use)			
Hex. Code	Octal Code	Graphic[†]	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	Graphic	Control Character[††]
00	000		NUL	00	000		null
01	001		SOH	01	001		start of header
02	002		STX	02	002		start of text
03	003		ETX	03	003		end of text
04	004		PF	20	040	space	
05	005		HT	09	011		horizontal tabulate
06	006		LC	20	040	space	
07	007		DEL	7F	177		delete
08	010		undefined	20	040	space	
09	011		undefined	20	040	space	
0A	012		SMM	20	040	space	
0B	013		VT	0B	013		vertical tabulate
0C	014		FF	0C	014		form feed
0D	015		CR	0D	015		carriage return
0E	016		SO	0E	016		shift out
0F	017		SI	0F	017		shift in
10	020		DLE	10	020		data link escape
11	021		DC1	11	021		device control 1
12	022		DC2	12	022		device control 2
13	023		TM	13	023		device control 3
14	024		RES	20	040	space	
15	025		NL	20	040	space	
16	026		BS	08	010		backspace
17	027		IL	20	040	space	
18	030		CAN	18	030		cancel
19	031		EM	19	031		end of medium
1A	032		CC	20	040	space	
1B	033		CU1	20	040	space	
1C	034		IFS	1C	034		file separator
1D	035		IGS	1D	035		group separator
1E	036		IRS	1E	036		record separator
1F	037		IUS	1F	037		unit separator
20	040		DS	20	040	space	
21	041		SOS	20	040	space	
22	042		FS	20	040	space	
23	043		undefined	20	040	space	
24	044		BYP	20	040	space	
25	045		LF	0A	012		linefeed
26	046		ETB or EOB	17	027		end of transmission block
27	047		ESC or PRE	1B	033		escape
28	050		undefined	20	040	space	
29	051		undefined	20	040	space	
2A	052		SM	20	040	space	
2B	053		CU2	20	040	space	
2C	054		undefined	20	040	space	
2D	055		ENQ	05	005		enquiry
2E	056		ACK	06	006		positive acknowledgment
2F	057		BEL	07	007		bell
30	060		undefined	20	040	space	
31	061		undefined	20	040	space	
32	062		SYN	16	026		synchronous idle
33	063		undefined	20	040	space	
34	064		PN	20	040	space	
35	065		RS	20	040	space	
36	066		UC	20	040	space	
37	067		EOT	04	004		end of transmission
38	070		undefined	20	040	space	
39	071		undefined	20	040	space	
3A	072		undefined	20	040	space	

Terminal EBCDIC				Network ASCII (Normalized Mode Use)			
Hex. Code	Octal Code	Graphic†	Control Character††	Hex. Code†††	Octal Code†††	Graphic	Control Character††
3B	073		CU3	20	040	space	
3C	074		DC4	14	024		device control 4
3D	075		NAK	15	025		negative acknowledgement
3E	076		undefined	20	040	space	
3F	077		SUB	1A	032		substitute
40	100	space		20	040	space	
41 thru 49	101 thru 111		undefined	20	040	space	
4A	112	¢		5B	133	[	
4B	113	.		2E	056	.	
4C	114	<		3C	074	<	
4D	115	(		28	050	(	
4E	116	+		2B	053	+	
4F	117	|		21	041	!	
50	120	&		26	046	&	
51 thru 59	121 thru 131		undefined	20	040	space	
5A	132	!		5D	135	]	
5B	133	$		24	044	$	
5C	134	*		2A	052	*	
5D	135	)		29	051	)	
5E	136	;		3B	073	;	
5F	137	¬		5E	136	∧	
60	140	-		2D	055	-	
61	141	/		2F	057	/	
62 thru 69	142 thru 151		undefined	20	040	space	
6A	152	¦		7C	174	¦	
6B	153	,		2C	054	,	
6C	154	%		25	045	%	
6D	155	_		5F	137	_	
6E	156	>		3E	076	>	
6F	157	?		3F	077	?	
70 thru 78	160 thru 170		undefined	20	040	space	
79	171	`		60	140	`	
7A	172	:		7A	172	:	
7B	173	#		23	043	#	
7C	174	@		40	100	@	
7D	175	'		27	047	'	
7E	176	=		3D	075	=	
7F	177	"		22	042	"	
80	200		undefined	20	040	space	
81	201	a		61	141	a	
82	202	b		62	142	b	
83	203	c		63	143	c	
84	204	d		64	144	d	
85	205	e		65	145	e	
86	206	f		66	146	f	
87	207	g		67	147	g	
88	210	h		68	150	h	
89	211	i		69	151	i	
8A thru 90	212 thru 220		undefined	20	040	space	

Terminal EBCDIC				Network ASCII (Normalized Mode Use)			
Hex. Code	Octal Code	Graphic[†]	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	Graphic	Control Character[††]
91	221	j		6A	152	j	
92	222	k		6B	153	k	
93	223	l		6C	154	l	
94	224	m		6D	155	m	
95	225	n		6E	156	n	
96	226	o		6F	157	o	
97	227	p		70	160	p	
98	230	q		71	161	q	
99	231	r		72	162	r	
9A thru A0	232 thru 240		undefined	20	040	space	
A1	241	~		7E	176	~	
A2	242	s		73	163	s	
A3	243	t		74	164	t	
A4	244	u		75	165	u	
A5	245	v		76	166	v	
A6	246	w		77	167	w	
A7	247	x		78	170	x	
A8	250	y		79	171	y	
A9	251	z		7A	172	z	
AA thru BF	252 thru 277		undefined	20	040	space	
C0	300	{		7B	173	{	
C1	301	A		41	101	A	
C2	302	B		42	102	B	
C3	303	C		43	103	C	
C4	304	D		44	104	D	
C5	305	E		45	105	E	
C6	306	F		46	106	F	
C7	307	G		47	107	G	
C8	310	H		48	110	H	
C9	311	I		49	111	I	
CA	312		undefined	20	040	space	
CB	313		undefined	20	040	space	
CC	314	♪		20	040	space	
CD	315		undefined	20	040	space	
CE	316	♀		20	040	space	
CF	317		undefined	20	040	space	
D0	320	}		7E	175	}	
D1	321	J		4A	112	J	
D2	322	K		4B	113	K	
D3	323	L		4C	114	L	
D4	324	M		4D	115	M	
D5	325	N		4E	116	N	
D6	326	O		4F	117	O	
D7	327	P		50	120	P	
D8	330	Q		51	121	Q	
D9	331	R		52	122	R	
DA thru DF	332 thru 337		undefined	20	040	space	
E0	340	\		5C	134	\	
E1	341		undefined	20	040	space	
E2	342	S		53	123	S	
E3	343	T		54	124	T	
E4	344	U		55	125	U	
E5	345	V		56	126	V	

Terminal EBCDIC				Network ASCII (Normalized Mode Use)			
Hex. Code	Octal Code	Graphic[†]	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	Graphic	Control Character[††]
E6	346	W		57	127	W	
E7	347	X		58	130	X	
E8	350	Y		59	131	Y	
E9	351	Z		5A	132	Z	
EA	352		undefined	20	040	space	
EB	353		undefined	20	040	space	
EC	354	⊓		20	040	space	
ED thru EF	355 thru 357		undefined	20	040	space	
F0	360	0		30	060	0	
F1	361	1		31	061	1	
F2	362	2		32	062	2	
F3	363	3		33	063	3	
F4	364	4		34	064	4	
F5	365	5		35	065	5	
F6	366	6		36	066	6	
F7	367	7		37	067	7	
F8	370	8		38	070	8	
F9	371	9		39	071	9	
FA	372			20	040	space	
FB thru FF	373 thru 377		undefined	20	040	space	

[†]Graphic characters shown are those used on the IBM System/370 standard (PN) print train.   Other devices support subsets or variations of this character graphic set.

[††]Not used for output to line printers.   Translation to a space (100 octal) occurs.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	ASCII Graphic	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character
00	000		NUL or ⓐ	00	000		null
03	003	▲	ETX or Ⓒ	03	003		end of text
05	005		ENQ or WRU or Ⓔ	05	005		enquiry
06	006		ACK or RU or Ⓕ	06	006		positive acknowledgement
09	011		HT or Ⓘ	09	011		horizontal tabulate
0A	012		LF or NL or ↓ or Ⓙ	0A	012		linefeed
0C	014		FF or FORM or Ⓛ	0C	014		formfeed
0F	017	≫	SI or Ⓞ	0F	017		shift in
11	021		DC1 or X-ON or Ⓠ	11	021		device control 1
12	022		DC2 or TAPE or Ⓡ	12	022		device control 2
14	024		DC4 or TAPE or Ⓣ	14	024		device control 4
17	027		ETB or Ⓦ	17	027		end transmission block
18	030		CAN or CLEAR or Ⓧ	18	030		cancel
1B	033		ESC or ESCAPE or Ⓘ	1B	033		escape
1D	035		GS or Ⓘ	1D	035		group separator
1E	036		RS or Ⓐ	1E	036		record separator
21	041	!		21	041	!	
22	042	"		22	042	"	
24	044	$		24	044	$	
27	047	'		27	047	'	
28	050	(		28	050	(	
2B	053	+		2B	053	+	
2D	055	−		2D	055	−	
2E	056	.		2E	056	.	
30	060	0		30	060	0	
33	063	3		33	063	3	
35	065	5		35	065	5	
36	066	6		36	066	6	
39	071	9		39	071	9	
3A	072	:		3A	072	:	
3C	074	<		3C	074	<	
3F	077	?		3F	077	?	
41	101	A		41	101	A	
42	102	B		42	102	B	
44	104	D		44	104	D	
47	107	G		47	107	G	
48	110	H		48	110	H	
4B	113	K		4B	113	K	
4D	115	M		4D	115	M	
4E	116	N		4E	116	N	
50	120	P		50	120	P	
53	123	S		53	123	S	
55	125	U		55	125	U	
56	126	V		56	126	V	
59	131	Y		59	131	Y	
5A	132	Z		5A	132	Z	
5C	134	\		5C	134	\	
5F	137	_ or ←		5F	137	_	
60	140	∽		60	140	∽	
63	143	c		63	143	c	
65	145	e		65	145	e	
66	146	f		66	146	f	
69	151	i		69	151	i	
6A	152	j		6A	152	j	
6C	154	l		6C	154	l	
6F	157	o		6F	157	o	
71	161	q		71	161	q	
72	162	r		72	162	r	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex. Code†	Octal Code†	ASCII Graphic	Control Character††	Hex. Code†††	Octal Code†††	ASCII Graphic	Control Character
74	164	t		74	164	t	
77	167	w		77	167	w	
78	170	x		78	170	x	
7B	173	{		7B	173	{	
7C	174	¦ or ↑ or │		7C	174	¦	
7D	175	}		7D	175	}	
7E	176	~ or ¬		7E	176	~	
81	201		SOH or Ⓐ	01	001		start of header
82	202		STX or Ⓑ	02	002		start of text
84	204		EOT or Ⓓ	04	004		end of transmission
87	207		BELL or Ⓖ	07	007		bell
88	210		BS or ← or Ⓗ	08	010		backspace
8B	213		VT or Ⓚ	0B	013		vertical tabulate
8D	215		CR or RETURN or Ⓜ	0D	015		carriage return
8E	216	≮	SO or Ⓝ	0E	016		shift out
90	220		DLE or Ⓟ	10	020		data link escape
93	223		DC3 or X-OFF or Ⓢ	13	023		device control 3
95	225		NAK or → or Ⓤ	15	025		negative acknowledgement
96	226		SYN or LINE CLEAR or Ⓥ	16	026		synchronous idle
99	231		EM or RESET or Ⓨ	19	031		end of medium
9A	232		SUB or ↑ or Ⓩ	1A	032		substitute
9C	234		FS or Ⓧ	1C	034		file separator
9F	237		US or Ⓞ	1F	037		unit separator
A0	240	SPACE or blank		20	040	space	
A3	243	#		23	043	#	
A5	245	%		25	045	%	
A6	246	&		26	046	&	
A9	251	)		29	051	)	
AA	252	*		2A	052	*	
AC	254	,		2C	054	,	
AF	257	/		2F	057	/	
B1	261	1		31	061	1	
B2	262	2		32	062	2	
B4	264	4		34	064	4	
B7	267	7		37	067	7	
B8	270	8		38	070	8	
BB	273	;		3B	073	;	
BD	275	=		3D	075	=	
BE	276	>		3E	076	>	
C0	300	@		40	100	@	
C3	303	C		43	103	C	
C5	305	E		45	105	E	
C6	306	F		46	106	F	
C9	311	I		49	111	I	
CA	312	J		4A	112	J	
CC	314	L		4C	114	L	
CF	317	O		4F	117	O	
D1	321	Q		51	121	Q	
D2	322	R		52	122	R	
D4	324	T		54	124	T	
D7	327	W		57	127	W	
D8	330	X		58	130	X	
DB	333	[		5B	133	[	
DD	335	]		5D	135	]	
DE	336	∧ or ¬		5E	136	∧	
E1	341	a		61	141	a	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	ASCII Graphic	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character
E2	342	b		62	142	b	
E4	344	d		64	144	d	
E7	347	g		67	147	g	
E8	350	h		68	150	h	
EB	353	k		6B	153	k	
ED	355	m		6D	155	m	
EE	356	n		6E	156	n	
F0	360	p		70	160	p	
F3	363	s		73	163	s	
F5	365	u		75	165	u	
F6	366	v		76	166	v	
F9	371	y		79	171	y	
FA	372	z		7A	172	z	
FF	377	▪	DEL or RUBOUT	7F	177		delete

[†]Shown with even parity, which is the default for these terminal classes (unless PA=N, an application program receives the same code as in normalized mode).

[††]A circle around a character indicates that the character key is pressed in conjunction with a CTL, CTRL, CNTRL, or CONTROL key to generate the code.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

TABLE A-10. CHARACTER CODE TRANSLATIONS, APL TYPEWRITER-PARING CONSOLES IN
TERMINAL CLASSES 1 THROUGH 3 AND 5 THROUGH 8 (M33, 713, 721, X3.64, H2000, T4014, M40)

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex Code[†]	Octal Code[†]	ASCII-APL Graphic	Control Character[††]	Hex Code[†††]	Octal Code[†††]	ASCII-APL Graphic	Control Character
00	000		NUL or Ⓐ	00	000		null
03	003	▲	ETX or Ⓒ	03	003		end of text
05	005		ENQ or WRU or Ⓔ	05	005		enquiry
06	006		ACK or RU or Ⓕ	06	006		positive acknowledgement
09	011		HT or Ⓘ	09	011		horizontal tabulate
0A	012		LF or NL or ↓ or Ⓙ	0A	012		linefeed
0C	014		FF or FORM or Ⓛ	0C	014		formfeed
0F	017	≫	SI or Ⓞ	0F	017		shift in
11	021		DC1 or X-ON or Ⓠ	11	021		device control 1
12	022		DC2 or TAPE or Ⓡ	12	022		device control 2
14	024		DC4 or TAPE or Ⓣ	14	024		device control 4
17	027		ETB or Ⓦ	17	027		end transmission block
18	030		CAN or CLEAR or Ⓧ	18	030		cancel
1B	033		ESC or ESCAPE or Ⓘ	1B	033		escape
1D	035		GS or Ⓙ	1D	035		group separator
1E	036		RS or Ⓐ	1E	036		record separator
21	041	.		23	043	.	
22	042	;		29	052	;	
24	044	<		40	100	<	
27	047	⊤		5D	135	⊤	
28	050	∨		21	041	∨	
2B	053	+		25	045	+	
2D	055	÷		2B	053	÷	
2E	056	.		2E	056	.	
30	060	0		30	060	0	
33	063	3		33	063	3	
35	065	5		35	065	5	
36	066	6		36	066	6	
39	071	9		39	071	9	
3A	072	(		28	050	(	
3C	074	;		3B	073	;	
3F	077	\		5C	134	\	
41	101	∝		61	141	∝	
42	102	⊥		62	142	⊥	
44	104	L		64	144	L	
47	107	∇		67	147	∇	
48	110	Δ		68	150	Δ	
4B	113	`		27	047	`	
4D	115	¦		6D	155	¦	
4E	116	⊤		6E	156	⊤	
50	120	*		2A	052	*	
53	123	⌈		73	163	⌈	
55	125	↓		75	165	↓	
56	126	U		76	166	U	
59	131	↑		79	171	↑	
5A	132	⊂		7A	172	⊂	
5C	134	⊢		7E	176	⊢	
5F	137	—		2D	055	—	
60	140	◇		60	140	◇	
63	143	C		43	103	C	
65	145	E		45	105	E	
66	146	F		46	106	F	
69	151	I		47	111	I	
6A	152	J		4A	112	J	
6C	154	L		4C	114	L	
6F	157	O		4F	117	O	
71	161	Q		51	121	Q	
72	162	R		52	122	R	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex Code†	Octal Code†	ASCII-APL Graphic	Control Character††	Hex Code†††	Octal Code†††	ASCII-APL Graphic	Control Character
74	164	T		54	124	T	
77	167	W		57	127	W	
78	170	X		58	130	X	
7B	173	{		7B	173	{	
7C	174	⊣		6B	153	⊣	
7D	175	}		7D	175	}	
7E	176	$		24	044	$	
81	201		SOH or Ⓐ	01	001		start of header
82	202		STX or Ⓑ	02	002		start of text
84	204		EOT or Ⓓ	04	004		end of transmission
87	207		BELL or Ⓖ	07	007		bell
88	210		BS or ← or Ⓗ	08	010		backspace
8B	213		VT or Ⓚ	0B	013		vertical tabulate
8D	215		CR or RETURN or Ⓜ	0D	015		carriage return
8E	216	≪	SO or Ⓝ	0E	016	≪	shift out
90	220		DLE or Ⓟ	10	020		data link escape
93	223		DC3 or X-OFF or Ⓢ	13	023		device control 3
95	225		NAK or → or Ⓤ	15	025		negative acknowledgement
96	226		SYN or LINE CLEAR or Ⓥ	16	026		synchronous idle
99	231		EM or RESET or Ⓨ	19	031		end of medium
9A	232		SUB or ↑ or Ⓩ	1A	032		substitute
9C	234		FS or Ⓝ	1C	034		file separator
9F	237		US or Ⓞ	1F	037		unit separator
A0	240	SPACE or blank		20	040	space	
A3	243	<		3C	074	<	
A5	245	=		3D	075	=	
A6	246	>		3E	076	>	
A9	251	∧		26	046	∧	
AA	252	≠		22	042	≠	
AC	254	,		2C	054	,	
AF	257	/		2F	057	/	
B1	261	1		31	061	1	
B2	262	2		32	062	2	
B4	264	4		34	064	4	
B7	267	7		37	067	7	
B8	270	8		38	070	8	
BB	273	[		5B	133	[	
BD	275	χ		66	146	χ	
BE	276	:		3A	072	:	
C0	300	‾		5E	136	‾	
C3	303	∩		63	143	∩	
C5	305	ε		65	145	ε	
C6	306			5F	137		
C9	311	⍳		69	151	⍳	
CA	312	∘		6A	152	∘	
CC	314	□		6C	154	□	
CF	317	○		6F	157	○	
D1	321	?		3F	077	?	
D2	322	ρ		72	162	ρ	
D4	324	~		74	164	~	
D7	327	ω		77	167	ω	
D8	330	⊃		78	170	⊃	
DB	333	←		70	160	←	
DD	335	→		71	161	→	
DE	336	≥		7C	174	≥	
E1	341	Ā		41	101	Ā	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex Code[†]	Octal Code[†]	ASCII-APL Graphic	Control Character[††]	Hex Code[†††]	Octal Code[†††]	ASCII-APL Graphic	Control Character
E2	342	B		42	102	B	
E4	344	D		44	104	D	
E7	347	G		47	107	G	
E8	350	H		48	110	H	
EB	353	K		4B	113	K	
ED	355	M		4D	115	M	
EE	356	N		4E	116	N	
F0	360	P		50	120	P	
F3	363	S		53	123	S	
F5	365	U		55	125	U	
F6	366	V		56	126	V	
F9	371	Y		59	131	Y	
FA	372	Z		5A	132	Z	
FF	377	■	DEL or RUBOUT	7F	177		delete

[†]Shown with even parity, which is the default for these terminal classes (unless PA=N, an application program receives the same code as in normalized mode).

[††]A circle around a character indicates that the character key is pressed in conjunction with a CTL, CTRL, CNTRL, or CONTROL key to generate the code.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

TABLE A-11. CHARACTER CODE TRANSLATIONS, APL BIT-PARING CONSOLES IN TERMINAL
CLASSES 1 THROUGH 3 AND 5 THROUGH 8 (M33, 713, 721, X3.64, H2000, T4014, AND M40)

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex Code[†]	Octal Code[†]	ASCII-APL Graphic	Control Character[††]	Hex Code[†††]	Octal Code[†††]	ASCII-APL Graphic	Control Character
00	000		NUL or Ⓐ	00	000		null
03	003	▲	ETX or Ⓒ	03	003		end of text
05	005		ENQ or WRU or Ⓔ	05	005		enquiry
06	006		ACK or RU or Ⓕ	06	006		positive acknowledgement
09	011		HT or Ⓘ	09	011		horizontal tabulate
0A	012		LF or NL or ↓ or Ⓙ	0A	012		linefeed
0C	014		FF or FORM or Ⓛ	0C	014		formfeed
0F	017	≫	SI or Ⓞ	0F	017		shift in
11	021		DC1 or X-ON or Ⓠ	11	021		device control 1
12	022		DC2 or TAPE or Ⓡ	12	022		device control 2
14	024		DC4 or TAPE or Ⓣ	14	024		device control 4
17	027		ETB or Ⓦ	17	027		end transmission block
18	030		CAN or CLEAR or Ⓧ	18	030		cancel
1B	033		ESC or ESCAPE or Ⓛ	1B	033		escape
1D	035		GS or Ⓙ	1D	035		group separator
1E	036		RS or Ⓐ	1E	036		record separator
21	041	¨		23	043	¨	
22	042	‾		5E	136	‾	
24	044	≤		40	100	≤	
27	047	≥		3E	076	≥	
28	050	≠		22	042	≠	
2B	053	(		28	050	(	
2D	055	+		2B	053	+	
2E	056	.		2E	056	.	
30	060	0		30	060	0	
33	063	3		33	063	3	
35	065	5		35	065	5	
36	066	6		36	066	6	
39	071	9		39	071	9	
3A	072	]		5D	135	]	
3C	074	;		3B	073	;	
3F	077	\		5C	134	\	
41	101	∝		61	141	∝	
42	102	⊥		62	142	⊥	
44	104	∟		64	144	∟	
47	107	∇		67	147	∇	
48	110	Δ		68	150	Δ	
4B	113	`		27	047	`	
4D	115				6D	155			
4E	116	T		6E	156	T	
50	120	*		2A	052	*	
53	123	⌈		73	163	⌈	
55	125	↓		75	165	↓	
56	126	∪		76	166	∪	
59	131	↑		79	171	↑	
5A	132	⊂		7A	172	⊂	
5C	134	◇		60	140	◇	
5F	137	∧		26	046	∧	
60	140	→		71	161	→	
63	143	C		43	103	C	
65	145	E		45	105	E	
66	146	F		46	106	F	
69	151	I		49	111	I	
6A	152	J		4A	112	J	
6C	154	L		4C	114	L	
6F	157	O		4F	117	O	
71	161	Q		51	121	Q	
72	162	R		52	122	R	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex Code†	Octal Code†	ASCII-APL Graphic	Control Character††	Hex Code†††	Octal Code†††	ASCII-APL Graphic	Control Character
74	164	T		54	124	T	
77	167	W		57	127	W	
78	170	X		58	130	X	
7B	173	⊣		6B	153	⊣	
7C	174	$		24	044	$	
7D	175	}		7D	160	}	
7E	176	÷		25	045	÷	
81	201		SOH or Ⓐ	01	001		start of header
82	202		STX or Ⓑ	02	002		start of text
84	204		EOT or Ⓓ	04	004		end of transmission
87	207		BELL or Ⓖ	07	007		bell
88	210		BS or ← or Ⓗ	08	010		backspace
8B	213		VT or Ⓚ	0B	013		vertical tabulate
8D	215		CR or RETURN or Ⓜ	0D	015		carriage return
8E	216	≪	SO or Ⓝ	0E	016		shift out
90	220		DLE or Ⓟ	10	020		data link escape
93	223		DC3 or X-OFF or Ⓢ	13	023		device control 3
95	225		NAK or → or Ⓤ	15	025		negative acknowledgement
96	226		SYN or LINE CLEAR or Ⓥ	16	026		synchronous idle
99	231		EM or RESET or Ⓨ	19	031		end of medium
9A	232		SUB or ↑ or Ⓩ	1A	032		substitute
9C	234		FS or Ⓧ̸	1C	034		file separator
9F	237		US or ⊖	1F	037		unit separator
A0	240	SPACE or blank		20	040	space	
A3	243	<		3C	074	<	
A5	245	=		3D	075	=	
A6	246	≥		7C	174	≥	
A9	251	∨		21	041	∨	
AA	252	)		29	051	)	
AC	254	,		2C	054	,	
AF	257	/		2F	057	/	
B1	261	1		31	061	1	
B2	262	2		32	062	2	
B4	264	4		34	064	4	
B7	267	7		37	067	7	
B8	270	8		38	070	8	
BB	273	[		5B	133	[	
BD	275	–		2D	055	–	
BE	276	:		3A	072	:	
C0	300	←		70	160	←	
C3	303	∩		63	143	∩	
C5	305	∈		65	145	∈	
C6	306	_		5F	137	_	
C9	311	⍳̃		69	151	⍳̃	
CA	312	∘		6A	152	∘	
CC	314	□		6C	154	□	
CF	317	○		6F	157	○	
D1	321	?		3F	077	?	
D2	322	ρ		72	162	ρ	
D4	324	~		74	164	~	
D7	327	ω		77	167	ω	
D8	330	⊃		78	170	⊃	
DB	333	⊢		7E	176	⊢	
DD	335	{		7B	173	{	
DE	336	χ		66	146	χ	
E1	341	A		41	101	A	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex Code[†]	Octal Code[†]	ASCII–APL Graphic	Control Character[††]	Hex Code[†††]	Octal Code[†††]	ASCII–APL Graphic	Control Character
E2	342	B		42	102	B	
E4	344	D		44	104	D	
E7	347	G		47	107	G	
E8	350	H		48	110	H	
EB	353	K		4B	113	K	
ED	355	M		4D	115	M	
EE	356	N		4E	116	N	
F0	360	P		50	120	P	
F3	363	S		53	123	S	
F5	365	U		55	125	U	
F6	366	V		56	126	V	
F9	371	Y		59	131	Y	
FA	372	Z		5A	132	Z	
FF	377	▪	DEL or RUBOUT	7F	177		delete

[†]Shown with even parity, which is the default for these terminal classes (unless PA=N, an application program receives the same code as in normalized mode).

[††]A circle around a character indicates that the character key is pressed in conjunction with a CTL, CTRL, CNTRL, or CONTROL key to generate the code.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

TABLE A-12. CHARACTER CODE TRANSLATIONS, ASCII CONSOLES AND LINE PRINTERS IN
TERMINAL CLASSES 10 AND 15 (200UT AND 734)

Terminal ASCII[†]				Network ASCII (Normalized Mode Use)				
Hex. Code[††]	Octal Code[††]	Keyboard or Printer Graphic		Input or Output		Console Output Only		Graphic
		ASCII	CDC	Hex. Code[†††]	Octal Code[†††]	Hex. Code[†††]	Octal Code[†††]	
20	040	blank	blank	20	040			space
23	043	#		23	043			#
25	045	%	%	25	045			%
26	046	&		26	046			&
29	051	)	)	29	051			)
2A	052	*	*	2A	052			*
2C	054	,	,	2C	054			,
2F	057	/	/	2F	057			/
31	061	1	1	31	061			1
32	062	2	2	32	062			2
34	064	4	4	34	064			4
37	067	7	7	37	067			7
38	070	8	8	38	070			8
3B	073	;	;	3B	073			;
3D	075	=	=	3D	075			=
3E	076	>	>	3E	076			>
40	100	@	≤	40	100	60	140	@
43	103	C	C	43	103	63	143	C
45	105	E	E	45	105	65	145	E
46	106	F	F	46	106	66	146	F
49	111	I	I	49	111	69	151	I
4A	112	J	J	4A	112	6A	152	J
4C	114	L	L	4C	114	6C	154	L
4F	117	O	O	4F	117	6F	157	O
51	121	Q	Q	51	121	71	161	Q
52	122	R	R	52	122	72	162	R
54	124	T	T	54	124	74	164	T
57	127	W	W	57	127	77	167	W
58	130	X	X	58	130	78	170	X
5B	133	[	[	5B	133	7B	173	[
5D	135	]	]	5D	135	7D	175	]
5E	136	∧	¬	5E	136	7E	176	∧
A1	241	!		21	041			!
A2	242	"	≠	22	042			"
A4	244	$	$	24	044			$
A7	247	'		27	047			'
A8	250	(	(	28	050			(

Terminal ASCII[†]				Network ASCII (Normalized Mode Use)				
Hex. Code[††]	Octal Code[††]	Keyboard or Printer Graphic		Input or Output		Console Output Only		Graphic
		ASCII	CDC	Hex. Code[†††]	Octal Code[†††]	Hex. Code[†††]	Octal Code[†††]	
AB	253	+	+	2B	053			+
AD	255	−	−	2D	055			−
AE	256	.	..	2E	056			.
B0	260	0	0	30	060			0
B3	263	3	3	33	063			3
B5	265	5	5	35	065			5
B6	266	6	6	36	066			6
B9	271	9	9	39	071			9
BA	272	:	:	3A	072			:
BC	274	<	<	3C	074			<
BF	277	?	↓	3F	077			?
C1	301	A	A	41	101	61	141	A
C2	302	B	B	42	102	62	142	B
C4	304	D	D	44	104	64	144	D
C7	307	G	G	47	107	67	147	G
C8	310	H	H	48	110	68	150	H
CB	313	K	K	4B	113	6B	153	K
CD	315	M	M	4D	115	6D	155	M
CE	316	N	N	4E	116	6E	156	N
D0	320	P	P	50	120	70	160	P
D3	323	S	S	53	123	73	163	S
D5	325	U	U	55	125	75	165	U
D6	326	V	V	56	126	76	166	V
D9	331	Y	Y	59	131	79	171	Y
DA	332	Z	Z	5A	132	7A	172	Z
DC	334	\	≥	5C	134	7C	174	\
DF	337	_	↦	5E	135	7F	177	_

[†]Escape codes are not listed.

[††]Shown with odd parity, the only possible parity selection for these terminal classes. ASCII control codes 000 through 040$_8$ (without parity) are removed from input during complete editing; codes 01$_8$ and 03$_8$ (SOH and ETX, without parity) are preserved as data in full-ASCII mode, as are escape code sequences.

[†††]Shown with zero parity (eighth or uppermost bit is always zero). During output, codes 000 through 010$_8$ are converted to code 040$_8$ (blank); codes 012$_8$, 015$_8$, and 037$_8$ (LF, CR, and US) are removed. Codes for lowercase ASCII characters sent to the console are converted to the codes for the equivalent uppercase characters supported by the terminal, as shown.

TABLE A-13. CHARACTER CODE TRANSLATIONS, EXTERNAL BINARY CODED (BCD) CONSOLES AND LINE PRINTERS IN TERMINAL CLASSES 10 AND 15 (200UT AND 734)

Terminal External BCD[†]		Keyboard or Printer Graphic		Network ASCII (Normalized Mode Use)				Graphic
Hex. Code[††]	Octal Code[††]			Input or Output		Console Output Only		
		ASCII	CDC	Hex. Code[†††]	Octal Code[†††]	Hex. Code[†††]	Octal Code[†††]	
10	020	:	:	3A	072			:
20	040	–	–	2D	055			–
23	043	L	L	4C	114	6C	154	L
25	045	N	N	4E	116	6E	156	N
26	046	O	O	4F	117	6F	157	O
29	051	R	R	52	122	72	162	R
2A	052	!	∨	21	041			!
2C	054	*	*	2A	052			*
2F	057	>	>	3E	076			>
31	061	A	A	41	101	61	141	A
32	062	B	B	42	102	62	142	B
34	064	D	D	44	104	64	144	D
37	067	G	G	47	107	67	147	G
38	070	H	H	48	110	68	150	H
3B	073	.	.	2E	056			.
3D	075			5C	134	7C	174	\
43	103	3	3	33	063			3
45	105	5	5	35	065			5
46	106	6	6	36	066			6
49	111	9	9	39	071			9
4A	112	0	0	30	060			0
4C	114	=	≠	22	042			"
4F	117	[	[	5B	133	7B	173	[
51	121	/	/	2F	057			/
52	122	S	S	53	123	73	163	S
54	124	U	U	55	125	75	165	U
57	127	X	X	58	130	78	170	X
58	130	Y	Y	59	131	79	171	Y
5B	133	,	,	2C	054			,
5D	135	_	⟶	5F	137	7F	177	_
5E	136	#	≡	23	043			#
A1	241	J	J	4A	112	6A	152	J
A2	242	K	K	4B	113	6B	153	K
A4	244	M	M	4D	115	6D	155	M
A7	247	P	P	50	120	70	160	P
A8	250	Q	Q	51	121	71	161	Q
AB	253	$	$	24	044			$

Terminal External BCD[†]				Network ASCII (Normalized Mode Use)				
Hex. Code[††]	Octal Code[††]	Keyboard or Printer Graphic		Input or Output		Console Output Only		Graphic
		ASCII	CDC	Hex. Code[†††]	Octal Code[†††]	Hex. Code[†††]	Octal Code[†††]	
AD	255	'	↑	27	047			,
AE	256	?	↓	3F	077			?
B3	263	C	C	43	103	63	143	C
B5	265	E	E	45	105	65	145	E
B6	266	F	F	46	106	66	146	F
B9	271	I	I	49	111	69	151	I
BA	272	<	<	3C	074			<
BC	274	)	)	29	.051			)
BF	277	;	;	3B	073			;
C1	301	1	1	31	061			1
C2	302	2	2	32	062			2
C4	304	4	4	34	064			4
C7	307	7	7	37	067			7
C8	310	8	8	38	070			8
CB	313	=	=	3D	075			=
CD	315	@	≤	40	100	60	140	@
CE	316	%	%	25	045			%
D0	320	blank	blank	20	040			space
D3	323	T	T	54	124	74	164	T
D5	325	V	V	56	126	76	166	V
D6	326	W	W	57	127	77	167	W
D9	331	Z	Z	5A	132	7A	172	Z
DA	332	]	]	5D	135	7D	175	]
DC	334	(	(	28	050			(
DF	337	&	∧	26	046			&
D0	320	∧ or blank	¬ or ■ or none			5E, 7E	136, 176	∧[§]

[†]Escape codes and control codes are not listed.

[††]Shown with odd parity, the only possible parity selection for these terminal classes.

[†††]Shown with zero parity (eighth or uppermost bit is always zero). During output, codes 000 through 037$_8$ are converted to code 320$_8$ (blank). Codes for lowercase ASCII characters sent to the console are converted to the codes for the equivalent uppercase characters supported by the terminal, as shown.

[§]Input and output of this symbol is not possible on some terminals. BCD transmission conventions support the rubout symbol ■ as an internal terminal memory parity error indicator instead. The ASCII codes 136$_8$ and 176$_8$ are output as a blank.

TABLE A-14. CHARACTER CODE TRANSLATIONS, CONSOLES AND LINE PRINTERS IN TERMINAL CLASSES 11, 12 AND 13 (711, 714, AND 714X)

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	ASCII Graphic	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character[§]
01	001		SOH or Ⓐ	01	001		start of header[§§]
02	002		STX or Ⓑ	20	040	space	
04	004		EOT or Ⓓ	20	040	space	
07	007		BELL or Ⓖ	20	040	space	
08	010		BS or ← or Ⓗ	20	040	space	
0B	013		VT or Ⓚ	0B	013		vertical tabulate
0D	015		CR or RETURN or Ⓜ				
0E	016		SO or Ⓝ	0E	016		shift out
10	020		DLE or Ⓟ	10	020		data link escape
13	023		DC3 or X-OFF or Ⓢ	13	023		device control 3
15	025		NAK or → or Ⓤ	15	025		negative acknowledgment
16	026		SYN or LINE CLEAR or Ⓥ	16	026		synchronous idle
19	031		EM or RESET or Ⓨ	19	031		end of medium
1A	032		SUB or ↑ or Ⓩ	1A	032		substitute
1C	034		FS or Ⓧ̸	1C	034		file separator
1F	037		US or ⊖	20	040	space	
20	040	SPACE or blank		20	040	space	
23	043	#		23	043	#	
25	045	%		25	045	%	
26	046	&		26	046	&	
29	051	)		29	051	)	
2A	052	*		2A	052	*	
2C	054	,		2C	054	,	
2F	057	/		2F	057	/	
31	061	1		31	061	1	
32	062	2		32	062	2	
34	064	4		34	064	4	
37	067	7		37	067	7	
38	070	8		38	070	8	
3B	073	;		3B	073	;	
3D	075	=		3D	075	=	
3E	076	>		3E	076	>	
40	100	@		40	100	@	
43	103	C		43	103	C	
45	105	E		45	105	E	
46	106	F		46	106	F	
49	111	I		49	111	I	
4A	112	J		4A	112	J	
4C	114	L		4C	114	L	
4F	117	O		4F	117	O	
51	121	Q		51	121	Q	
52	122	R		52	122	R	
54	124	T		54	124	T	
57	127	W		57	127	W	
58	130	X		58	130	X	
5B	133	[		5B	133	[	
5D	135	]		5D	135	]	
5E	136	∧ or ¬		5E	136	∧	
61	141	a		61	141	a	
62	142	b		62	142	b	
64	144	d		64	144	d	
67	147	g		67	147	g	
68	150	h		68	150	h	
6B	153	k		6B	153	k	
6D	155	m		6D	155	m	
6E	156	n		6E	156	n	
70	160	p		70	160	p	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	ASCII Graphic	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character[§]
73	163	s		73	163	s	
75	165	u		75	165	u	
76	166	v		76	166	v	
79	171	y		79	171	y	
7A	172	z		7A	172	z	
7C	174	¦ or ↑ or │		7C	174	¦	
7F	177	■	DEL or RUBOUT	7F	177		delete
80	200		NUL or Ⓐ	20	040	space	
83	203		ETX or Ⓒ	03	003		end of text[§§]
85	205		ENQ or WRU or Ⓔ	20	040	space	
86	206		ACK or RU or Ⓕ	20	040	space	
89	211		HT or Ⓘ	09	011		horizontal tabulate
8A	212		LF or NL or ↓ or Ⓙ or NEW LINE	0A	012		linefeed
8C	214		FF or FORM or Ⓛ	0C	014		formfeed
8F	217		SI or Ⓞ	0F	017		shift in
91	221		DC1 or X-ON or Ⓠ	11	021		device control 1
92	222		DC2 or TAPE or Ⓡ	12	022		device control 2
94	224		DC4 or TAPE or Ⓣ	14	024		device control 4
97	227		ETB or Ⓦ	17	027		end transmission block
98	230		CAN or CLEAR or Ⓧ	18	030		cancel
9B	233		ESC or ESCAPE or Ⓘ	1B	033		escape
9D	235		GS or Ⓙ	1D	035		group separator
9E	236		RS or Ⓐ	1E	036		record separator
A1	241	!		21	041	!	
A2	242	"		22	042	"	
A4	244	$		24	044	$	
A7	247	'		27	047	'	
A8	250	(		28	050	(	
AB	253	+		2B	053	+	
AD	255	-		2D	055	-	
AE	256	.		2E	056	.	
B0	260	0		30	060	0	
B3	263	3		33	063	3	
B5	265	5		35	065	5	
B6	266	6		36	066	6	
B9	271	9		39	071	9	
BA	272	:		3A	072	:	
BC	274	<		3C	074	<	
BF	277	?		3F	077	?	
C1	301	A		41	101	A	
C2	302	B		42	102	B	
C4	304	D		44	104	D	
C7	307	G		47	107	G	
C8	310	H		48	110	H	
CB	313	K		4B	113	K	
CD	315	M		4D	115	M	
CE	316	N		4E	116	N	
D0	320	P		50	120	P	
D3	323	S		53	123	S	
D5	325	U		55	125	U	
D6	326	V		56	126	V	
D9	331	Y		59	131	Y	
DA	332	Z		5A	132	Z	
DC	334	\		5C	134	∨	
DF	337	⎯ or ←		5F	137	⎯	
E0	340	ⸯ		60	140	ⸯ	
E3	343	c		63	143	c	

Terminal ASCII (Transparent Mode Use)				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	ASCII Graphic	Control Character[††]	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character[§]
E5	345	e		65	145	e	
E6	346	f		66	146	f	
E9	351	i		69	151	i	
EA	352	j		6A	152	j	
EC	354	l		6C	154	l	
EF	357	o		6F	157	o	
F1	361	q		71	161	q	
F2	362	r		72	162	r	
F4	364	t		74	164	t	
F7	367	w		77	167	w	
F8	370	x		78	170	x	
FB	373	{		7B	173	{	
FD	375	}		7D	175	}	
FE	376	~ or ⌐		7E	176	~	

[†]Shown with odd parity, which is the default for these terminal classes (unless PA=N, an application program receives the same code as in normalized mode).

[††]A circle around a character indicates that the character key is pressed in conjunction with a CTL, CTRL, CNTRL, or CONTROL key to generate the code.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

[§]Converted to a space (040_8) within a batch printer file.

[§§]Converted to a space (040_8) during complete editing.

Terminal EBCD				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	EBCD Graphic[††]	Control Character	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character
01	001	_ or –		5F or 2D	137 or 055	_ or –	
02	002	⌀ or @		21 or 40	140 or 100	⌐ or @	
04	004	* or 8		2A or 38	052 or 070	* or 8	
07	007	H or h		48 or 68	110 or 150	H or h	
08	010	: or 4		3A or 34	072 or 064	: or 4	
0B	013	D or d		44 or 64	104 or 144	D or d	
0D	015		RES or RESTORE	00	000		null
0E	016		BY or BYPASS	00	000		null
10	020	< or 2		3C or 32	074 or 062	< or 2	
13	023	B or b		42 or 62	102 or 142	B or b	
15	025		undefined	00	000		null
16	026		undefined	00	000		null
19	031	O or o		4F or 6F	117 or 157	O or o	
1A	032	W or w		57 or 77	127 or 167	W or w	
1C	034		UCS or UPPERCASE	0E	016		shift out[§]
1F	037		LCS or LOWERCASE	0F	017		shift in[§]
20	040	= or 1		3D or 31	075 or 061	= or 1	
23	043	A or a		41 or 61	101 or 141	A or a	
25	045	R or r		52 or 72	122 or 162	R or r	
26	046	Z or z		5A or 7A	132 or 172	Z or z	
29	051	N or n		4E or 6E	116 or 156	N or n	
2A	052	V or v		56 or 76	126 or 166	V or v	
2C	054		RO or READER STOP	14	024		device control 4
2F	057		HT or TAB	09	011		horizontal tabulate
31	061	L or l		4C or 6C	114 or 154	L or l	
32	062	T or t		54 or 74	124 or 164	T or t	
34	064	" or #		22 or 23	042 or 043	= or #	
37	067	¬ or .		5E or 2E	136 or 056	∧ or .	
38	070	> or 7		3E or 37	076 or 067	> or 7	
3B	073	G or g		47 or 67	107 or 147	G or g	
3D	075		IL or IDLE or NULL	00	000		null
3E	076		PRE or PREFIX	01	001		start of header[§]
40	100	space		20	040	space	
43	103	+ or &		2B or 26	053 or 046	+ or &	
45	105	Q or q		51 or 71	121 or 161	Q or q	
46	106	Y or y		59 or 79	131 or 171	Y or y	
49	111	M or m		4D or 6D	115 or 155	M or m	
4A	112	U or u		55 or 75	125 or 165	U or u	
4C	114		PN or PUNCH ON	11	021		device control 1 (tape on)
4F	117		PF or PUNCH OFF	13	023		device control 3 (tape off)
51	121	K or k		4B or 6B	113 or 153	K or k	
52	122	S or s		53 or 73	123 or 163	S or s	
54	124	) or 0		29 or 30	051 or 060	) or 0	
57	127		undefined	00	000		null
58	130	′ or 6		27 or 36	047 or 066	′ or 6	
5B	133	F or f		46 or 66	106 or 146	F or f	
5D	135		BS or BACKSPACE	08	010		backspace
5E	136		EOB	17	027		end transmission block[§]
61	141	J or j		4A or 6A	112 or 152	J or j	
62	142	? or /		3F or 2F	077 or 057	? or /	
64	144	(or 9		28 or 39	050 or 071	(or 9	
67	147	I or i		49 or 69	111 or 151	I or i	
68	150	% or 5		25 or 35	045 or 065	% or 5	
6B	153	E or e		45 or 65	105 or 145	E or e	
6D	155		NL or CR or RETURN	0D	015		carriage return
6E	156		LF or LINE FEED	0A	012		linefeed

Terminal EBCD				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	EBCD Graphic[††]	Control Character	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character
70	160	; or 3		3B or 33	073 or 063	; or 3	
73	163	C or c		43 or 63	103 or 143	C or c	
75	165	! or $		21 or 24	041 or 044	! or $	
76	166	¦ or ,		7C or 2C	174 or 054	¦ or ,	
79	171	P or p		50 or 70	120 or 160	P or p	
7A	172	X or x		58 or 78	130 or 170	X or x	
7C	174		EOT	04	004		end of transmission[§]
7F	177		DEL	7F	177		delete
00	000	space		5B thru 5D	133 thru 135	[or \ or]	
00	000	space		60	140	`	
00	000	space		7B	173	{	
00	000	space		7D or 7E	175 or 176	} or ~	
3D	075		IL or IDLE or NULL[§§]	02	002		start of text
3D	075		IL or IDLE or NULL[§§]	03	003		end of text
3D	075		IL or IDLE or NULL[§§]	05	005		enquire
3D	075		IL or IDLE or NULL[§§]	07	007		bell
3D	075		IL or IDLE or NULL[§§]	0B or 0C	013 or 014		vertical tabulate or formfeed
3D	075		IL or IDLE or NULL[§§]	10	020		data link escape
3D	075		IL or IDLE or NULL[§§]	12	022		device control 2
3D	075		IL or IDLE or NULL[§§]	14 thru 16	024 thru 026		device control 4, negative acknowledge, or synchronize
3D	075		IL or IDLE or NULL[§§]	18 thru 1F	030 thru 037		cancel, end of media, substitute, escape, file separator, group separator, record separator, or unit separator

[†]Shown with odd parity; odd parity is the default for this terminal class.

[††]Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

[§]Not transmitted to the host computer after translation during input.

[§§]Output translation only.

Terminal EBCD-APL				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	EBCD-APL Graphic[††]	Control Character	Hex Code[†††]	Octal Code[†††]	ASCII-APL Graphic	Control Character
01	001	_ or +		5F or 2D	137 or 053	_ or +	
02	002	→ or ←		71 or 70	161 or 160	→ or ←	
04	004	≠ or 8		22 or 38	042 or 070	≠ or 8	
07	007	Δ or H		68 or 48	150 or 110	Δ or H	
08	010	≤ or 4		40 or 34	100 or 064	≤ or 4	
0B	013	⌊ or D		64 or 44	144 or 104	⌊ or D	
0D	015		undefined	00	000		null
0E	016		undefined	00	000		null
10	020	- or 2		2D or 32	055 or 062	- or 2	
13	023	⊥ or B		42 or 62	142 or 102	⊥ or B	
15	025		undefined	00	000		null
16	026		undefined	00	000		null
19	031	○ or O		6F or 4F	157 or 117	○ or O	
1A	032	ω or W		77 or 57	167 or 127	ω or W	
1C	034		UCS or UPPERCASE	0E	016		shift out[§]
1F	037		LCS or LOWERCASE	0F	017		shift in[§]
20	040	" or 1		22 or 31	042 or 061	" or 1	
23	043	∝ or A		61 or 41	141 or 101	∝ or A	
25	045	ρ or R		72 or 52	162 or 122	ρ or R	
26	046	⊂ or Z		7A or 5A	172 or 132	⊂ or Z	
29	051	τ or N		6E or 4E	156 or 116	τ or N	
2A	052	∪ or V		76 or 56	166 or 126	∪ or V	
2C	054		undefined	00	000		null
2F	057		HT or TAB	06	006		horizontal tabulate
31	061	□ or L		6C or 4C	154 or 114	□ or L	
32	062	~ or T		74 or 54	164 or 124	~ or T	
34	064	) or]		29 or 5D	051 or 135	) or]	
37	067	: or .		3A or 2E	072 or 056	: or .	
38	070	> or 7		3E or 37	076 or 067	> or 7	
3B	073	∇ or G		67 or 47	147 or 107	∇ or G	
3D	075		IL or IDLE or NULL	00	000		null
3E	076		PRE or PREFIX	1B	033		escape
40	100	space		20	040	space	
43	103	+ or X		25 or 66	045 or 146	+ or X	
45	105	? or Q		3F or 51	077 or 121	? or Q	
46	106	↑ or Y		79 or 59	171 or 131	↑ or Y	
49	111	∣ or M		6D or 4D	155 or 115	∣ or M	
4A	112	↓ or U		75 or 55	165 or 125	↓ or U	
4C	114		undefined	00	000		null
4F	117		undefined	00	000		null
51	121	⊣ or K		6B or 4B	153 or 113	⊣ or K	
52	122	⌈ or S		73 or 53	163 or 123	⌈ or S	
54	124	∧ or 0		26 or 30	046 or 060	∧ or 0	
57	127		undefined	00	000		null
58	130	≥ or 6		7C or 36	174 or 066	≥ or 6	
5B	133	= or F		5E or 46	136 or 106	= or F	
5D	135		BS or BACKSPACE	08	010		backspace
5E	136		EOB	17	027		end transmission block[§]
61	141	○ or J		6A or 4A	152 or 112	○ or J	
62	142	\ or /		5C or 2F	134 or 057	\ or /	
64	144	∨ or 9		21 or 39	041 or 071	∨ or 9	
67	147	⍳ or I		69 or 49	151 or 111	⍳ or I	
68	150	= or 5		3D or 35	075 or 065	= or 5	
6B	153	∈ or E		65 or 45	145 or 105	∈ or E	
6D	155		NL or CR or RETURN	0D	015		carriage return
6E	156		LF or LINE FEED	0A	012		line feed
70	160	< or 3		3C or 33	074 or 063	< or 3	
73	163	∩ or C		63 or 43	143 or 103	∩ or C	
75	165	(or [		28 or 5B	050 or 133	(or [	
76	166	; or ,		3B or 2C	073 or 054	; or ,	
79	171	* or P		2A or 50	052 or 120	* or P	
7A	172	⊃ or X		78 or 58	170 or 130	⊃ or X	

Terminal EBCD-APL				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	EBCD-APL Graphic[††]	Control Character	Hex. Code[†††]	Octal Code[†††]	ASCII-APL Graphic	Control Character
7C	174		EOT	04	004		end of transmission§
7F	177		DEL	7F	177		delete
00	000	space§§		27	047	'	
00	000	space§§		60	140	◊	
00	000	space§§		7B	173	{	
00	000	space§§		7D	175	}	
3D	075		IL or IDLE or NULL§§	02	002		start of text
3D	075		IL or IDLE or NULL§§	03	003		end of text
3D	075		IL or IDLE or NULL§§	05	005		enquire
3D	075		IL or IDLE or NULL§§	07	007		bell
3D	075		IL or IDLE or NULL§§	0B or 0C	013 or 014		vertical tabulate or form feed
3D	075		IL or IDLE or NULL§§	10 thru 16	020 thru 026		data link escape, device control 1 thru device control 4, negative acknowledge, or synchronize
3D	075		IL or IDLE or NULL§§	18 thru 1F	030 thru 037		cancel, end of media, substitute, escape file separator, group separator, record separator, or unit separator

[†]Shown with odd parity; odd parity is the default for this terminal class.

[††]Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

§Not transmitted to the host computer after translation during input.

§§Output translation only.

TABLE A-17. ASCII CHARACTER CODE TRANSLATIONS, CORRESPONDENCE
CODE CONSOLES IN TERMINAL CLASS 4 (2741)

Terminal Correspondence Code				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	Correspondence Code Graphic[††]	Control Character	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character
01	001	1/4 or 1/2		5B or 5D	137 or 135	[or]	
02	002	T or t		54 or 74	124 or 164	T or t	
04	004	$ or 4		24 or 34	044 or 064	$ or 4	
07	007	? or /		3F or 2F	077 or 057	? or /	
08	010	% or 5		25 or 35	045 or 065	% or 5	
0B	013	P or p		50 or 70	120 or 160	P or p	
0D	015		RES or RESTORE	00	000		null
0E	016		BY or BYPASS	00	000		null
10	020	@ or 2		40 or 32	100 or 062	@ or 2	
13	023	+ or =		2B or 3D	053 or 075	+ or =	
15	025		undefined	00	000		null
16	026		undefined	00	000		null
19	031	I or i		49 or 69	111 or 151	I or i	
1A	032	K or k		4B or 6B	113 or 153	K or k	
1C	034		UCS or UPPERCASE	0E	016		shift out[§]
1F	037		LCS or LOWERCASE	0F	017		shift in[§]
20	040	+ or 1		7C or 31	174 or 061	¦ or 1	
23	043	G or g		47 or 67	107 or 147	G or g	
25	045	S or s		53 or 73	123 or 163	S or s	
26	046	H or h		48 or 68	110 or 150	H or h	
29	051	R or r		52 or 72	122 or 162	R or r	
2A	052	D or d		44 or 64	104 or 144	D or d	
2C	054		RO or READER STOP	14	024		device control 4
2F	057		HT or TAB	09	011		horizontal tabulate
31	061	V or v		56 or 76	126 or 166	V or v	
32	062	U or u		55 or 75	125 or 165	U or u	
34	064	(or 9		28 or 39	050 or 071	(or 9	
37	067	_ or -		5F or 2D	137 or 055	_ or -	
38	070	* or 8		2A or 38	052 or 070	* or 8	
3B	073	,		2C	054	,	
3D	075		IL or IDLE or NULL	00	000		null
3E	076		PRE or PREFIX	1B	033		escape
40	100	space		20	040	space	
43	103	J or j		4A or 6A	112 or 152	J or j	
45	105	O or o		4F or 6F	117 or 157	O or o	
46	106	L or l		4C or 6C	114 or 154	L or l	
49	111	" or '		22 or 27	042 or 041	" or '	
4A	112	E or e		45 or 65	105 or 145	E or e	
4C	114		PN or PUNCH ON	11	021		device control 1 (tape on)
4F	117		PF or PUNCH OFF	13	023		device control 3 (tape off)
51	121	.		2E	056	.	
52	122	N or n		4E or 6E	116 or 156	N or n	
54	124	Z or z		5A or 7A	132 or 172	Z or z	
57	127		undefined	00	000		null
58	130	¢ or 6		21 or 36	041 or 066	! or 6	
5B	133	Q or q		51 or 71	121 or 161	Q or q	
5D	135		BS or BACKSPACE	08	010		backspace
5E	136		EOB	17	027		end transmission block[§]
61	141	M or m		4D or 6D	115 or 155	M or m	
62	142	X or x		58 or 78	130 or 170	X or x	
64	144	) or 0		29 or 30	051 or 060	) or 0	
67	147	Y or y		79 or 59	131 or 171	Y or y	
68	150	& or 7		26 or 37	046 or 067	& or 7	
6B	153	: or ;		3A or 3B	072 or 073	: or ;	
6D	155		NL or CR or RETURN	0D	015		carriage return
6E	156		LF or LINE FEED	0A	012		line feed
70	160	# or 3		23 or 33	043 or 063	# or 3	
73	163	F of f		46 or 66	106 or 146	F or f	
75	165	W or w		57 or 77	127 or 167	W or w	

Terminal Correspondence Code				Network ASCII (Normalized Mode Use)			
Hex. Code[†]	Octal Code[†]	Correspondence Code Graphic[††]	Control Character	Hex. Code[†††]	Octal Code[†††]	ASCII Graphic	Control Character
76	166	B or b		42 or 62	102 or 142	B or b	
79	171	A or a		41 or 61	101 or 141	A or a	
7A	172	C or c		43 or 63	103 or 143	C or c	
7C	174		EOT	04	004		end of transmission[§]
7F	177		DEL	18	030		cancel
00	000	space[§§]		27	047	'	
00	000	space[§§]		5C	134	\	
00	000	space[§§]		5E	136	^	
00	000	space[§§]		60	140	`	
00	000	space[§§]		7B	173	{	
00	000	space[§§]		7D or 7E	175 or 176	} or ~	
3D	075		IL or IDLE or NULL[§§]	01	001		start of header
3D	075		IL or IDLE or NULL[§§]	02	002		start of text
3D	075		IL or IDLE or NULL[§§]	03	003		end of text
3D	075		IL or IDLE or NULL[§§]	05	005		enquire
3D	075		IL or IDLE or NULL[§§]	07	007		bell
3D	075		IL or IDLE or NULL[§§]	0B or 0C	013 or 014		vertical tabulate or form feed
3D	075		IL or IDLE or NULL[§§]	10	020		data link escape
3D	075		IL or IDLE or NULL[§§]	12	022		device control 2
3D	075		IL or IDLE or NULL[§§]	14 thru 16	024 thru 026		device control 4, negative acknowledge, or synchronize
3D	075		IL or IDLE or NULL[§§]	18 thru 1F	030 thru 037		cancel, end of media, substitute, file separator, group separator, record separator, or unit separator

[†]Shown with odd parity; odd parity is the default for this terminal class.

[††]Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

[§]Not transmitted to the host computer after translation during input.

[§§]Output translation only.

TABLE A-18. APL CHARACTER CODE TRANSLATIONS, CORRESPONDENCE
CODE CONSOLES IN TERMINAL CLASS 4 (2741)

Terminal Correspondence Code				Network ASCII (Normalized Mode Use)			
Hex Code†	Octal Code†	Correspondence Code APL Graphic††	Control Character	Hex Code†††	Octal Code†††	ASCII-APL Graphic	Control Character
01	001	→ or ←		71 or 70	161 or 160	→ or ←	
02	002	~ or T		74 or 54	164 or 124	~ or T	
04	004	≤ or 4		40 or 34	100 or 064	≤ or 4	
07	007	\ or /		5C or 2F	134 or 057	\ or /	
08	010	= or 5		3D or 35	075 or 065	= or 5	
0B	013	* or P		2A or 50	052 or 120	* or P	
0D	015		undefined	00	000		null
0E	016		undefined	00	000		null
10	020	— or 2		5E or 32	136 or 062	— or 2	
13	023	+ or χ		25 or 66	045 or 146	+ or χ	
15	025		undefined	00	000		null
16	026		undefined	00	000		null
19	031	⍳ or I		69 or 49	151 or 111	⍳ or I	
1A	032	′ or K		27 or 4B	153 or 113	′ or K	
1C	034		UCS or UPPERCASE	0E	016		shift out§
1F	037		LCS or LOWERCASE	0F	017		shift in§
20	040	" or 1		23 or 31	042 or 061	" or 1	
23	043	∇ or G		67 or 47	147 or 107	∇ or G	
25	045	⌈ or S		73 or 53	163 or 123	⌈ or S	
26	046	∆ or H		68 or 48	150 or 110	∆ or H	
29	051	ρ or R		72 or 52	162 or 122	ρ or R	
2A	052	⌊ or D		64 or 44	144 or 104	⌊ or D	
2C	054		undefined	00	000		null
2F	057		HT or TAB	09	011		horizontal tabulate
31	061	U or V		76 or 56	166 or 126	U or V	
32	062	↓ or U		75 or 55	165 or 125	↓ or U	
34	064	∨ or 9		21 or 39	041 or 071	∨ or 9	
37	067	− or +		2D or 2B	055 or 053	− or +	
38	070	≠ or 8		22 or 38	042 or 070	≠ or 8	
3B	073	; or ,		3B or 2C	073 or 054	; or ,	
3D	075		IL or IDLE or NULL	00	000		null
3E	076		PRE or PREFIX	1B	033		escape
40	100	space		20	040	space	
43	103	○ or J		6A or 4A	156 or 112	○ or J	
45	105	⊖ or O		6F or 4F	157 or 117	⊖ or O	
46	106	□ or L		6C or 4C	154 or 114	□ or L	
49	111	) or]		29 or 5D	051 or 035	) or]	
4A	112	∊ or E		65 or 45	145 or 105	∊ or E	
4C	114		undefined	00	000		null
4F	117		undefined	13	023		null
51	121	: or .		3A or 2E	072 or 056	: or .	
52	122	τ or N		6E or 4E	156 or 116	τ or N	
54	124	⊂ or Z		7A or 5A	172 or 132	⊂ or Z	
57	127		undefined	00	000		null
58	130	≥ or 6		7C or 36	174 or 066	≥ or 6	
5B	133	? or Q		3F or 51	077 or 121	? or Q	
5D	135		BS or BACKSPACE	08	010		backspace
5E	136		EOB	17	027		end transmission block§
61	141	∣ or M		6D or 4D	155 or 115	∣ or M	
62	142	⊃ or X		78 or 58	170 or 130	⊃ or X	
64	144	∧ or 0		26 or 30	045 or 060	∧ or 0	
67	147	↑ or Y		79 or 59	171 or 131	↑ or Y	
68	150	> or 7		3E or 37	076 or 067	> or 7	
6B	153	(or [		28 or 5B	050 or 133	(or [	
6D	155		NL or CR or RETURN	0D	015		carriage return
6E	156		LF or LINE FEED	0A	012		line feed
70	160	< or 3		3C or 33	074 or 063	< or 3	
73	163	_ or F		5F or 46	137 or 106	_ or F	
75	165	ω or W		77 or 57	167 or 127	ω or W	

TABLE A-18. APL CHARACTER CODE TRANSLATIONS, CORRESPONDENCE
CODE CONSOLES IN TERMINAL CLASS 4 (2741) (Contd)

Terminal Correspondence Code				Network ASCII (Normalized Mode Use)			
Hex Code[†]	Octal Code[†]	Correspondence Code APL Graphic[††]	Control Character	Hex Code[†††]	Octal Code[†††]	ASCII-APL Graphic	Control Character
76	166	⊥ or B		62 or 42	142 or 102	⊥ or B	
79	171	∝ or A		61 or 41	141 or 101	∝ or A	
7A	172	∩ or C		63 or 43	143 or 103	∩ or C	
7C	174		EOT	04 or 14	004		end of transmission[§]
7F	177		DEL	18	030		cancel
00	000	space[§§]		27	047	′	
00	000	space[§§]		60	140	◊	
00	000	space[§§]		7B	173	{	
00	000	space[§§]		7D or 7E	175 or 176	} or ⊢	
3D	075		IL or IDLE or NULL[§§]	01	001		start of header
3D	075		IL or IDLE or NULL[§§]	02	002		start of text
3D	075		IL or IDLE or NULL[§§]	03	003		end of text
3D	075		IL or IDLE or NULL[§§]	05	005		enquire
3D	075		IL or IDLE or NULL[§§]	07	007		bell
3D	075		IL or IDLE or NULL[§§]	0B or 0C	013 or 014		vertical tabulate or form feed
3D	075		IL or IDLE or NULL[§§]	10	020		data link escape
3D	075		IL or IDLE or NULL[§§]	12	022		device control 2
3D	075		IL or IDLE or NULL[§§]	14 thru 16	024 thru 026		device control 4, negative acknowledge, or synchronize
3D	075		IL or IDLE or NULL[§§]	18 thru 1F	030 thru 037		cancel, end of media, substitute, file separator, group separator, record separator, or unit separator

[†]Shown with odd parity; odd parity is the default for this terminal class. (Unless PA=N, the application program receives the same code as in normalized mode.)

[††]Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††]Shown with zero parity (eighth or uppermost bit is always zero).

[§]Not transmitted to the host computer after translation during input.

[§§]Output translation only.

TABLE A-19. FULL ASCII NORMALIZED MODE APL CHARACTER SET

|← 128-Character Set →|
|← 96-Character Subset →|
|← 64-Character Subset →|

b4 b3 b2 b1	ROW	COLUMN →	0 (000)	1 (001)	2 (010)	3 (011)	4 (100)	5 (101)	6 (110)	7 (111)
0 0 0 0	0		NUL 000	DLE 020	SP 040	0 060	≤ 100	P 120	◊ 140	← 160
0 0 0 1	1		SOH 001	DC1 021	·· or ∨ 041	1 061	A 101	Q 121	∝ 141	→ 161
0 0 1 0	2		STX 002	DC2 022	≠ 042	2 062	B 102	R 122	⊥ 142	ρ 162
0 0 1 1	3		ETX 003	DC3 023	: 043	3 063	C 103	S 123	∩ 143	⌈ 163
0 1 0 0	4		EOT 004	DC4 024	$ 044	4 064	D 104	T 124	⌊ 144	~ 164
0 1 0 1	5		ENQ 005	NAK 025	÷ 045	5 065	E 105	U 125	∈ 145	↓ 165
0 1 1 0	6		ACK 006	SYN 026	∧ 046	6 066	F 106	V 126	χ 146	∪ 166
0 1 1 1	7		BEL 007	ETB 027	' 047	7 067	G 107	W 127	∇ 147	ω 167
1 0 0 0	8		BS 010	CAN 030	(050	8 070	H 110	X 130	Δ 150	⊃ 170
1 0 0 1	9		HT 011	EM 031	) 051	9 071	I 111	Y 131	⍳ 151	↑ 171
1 0 1 0	A		LF 012	SUB 032	* 052	: 072	J 112	Z 132	∘ 152	⊂ 172
1 0 1 1	B		VT 013	ESC 033	+ 053	; 073	K 113	[133	⊣ 153	{ 173
1 1 0 0	C		FF 104	FS 034	, 054	< 074	L 114	\ 134	□ 154	≥ 174
1 1 0 1	D		CR 015	GS 035	- 055	= 075	M 115	] 135		155	} 175
1 1 1 0	E		SO 016	RS 036	. 056	> 076	N 116	— 136	⊤ 156	⊢ 176
1 1 1 1	F		SI 017	US 037	/ 057	? 077	O 117	_ 137	○ 157	DEL† 177

†The graphic 95-character subset does not include DEL; refer to Terminal Transmission Modes in the text.

LEGEND:

Numbers under characters are the octal values for the 7-bit character codes used within the network.

Two levels of error processing occur during an NDL processor run:

Compilation diagnostic processing

File access and dayfile message processing

The diagnostic processing performed during the run is documented by the error summary listing produced as part of the listing output file. The NDL job's output file contains an error summary list only if errors were detected. If you suppress listing file generation (by using the L=0 command option), the output file contains an error summary list only if fatal errors were detected; suppressing generation of the listing file suppresses all warning diagnostics.

If any error is detected, the NDL processor produces one of the diagnostic messages described in the following subsection. Statement processing is completed if feasible. If a fatal error is detected, no valid network definition files or file content summary listings are created, and the NDL processor aborts the job. After such an abort, control of the job transfers to the EXIT area in the command portion of the job image.

Informative messages are always placed in the job dayfile. In certain instances, a diagnostic-free job can be aborted because of file access problems; these cases also produce job dayfile messages. The dayfile messages are explained at the end of this appendix.

DIAGNOSTIC MESSAGES

The Network Definition Language job's error summary lists all errors that occurred in the program with a cross-reference to the line number of the statement unit in the source listing where the error occurred; the decimal line numbers are assigned sequentially from 1 through 100000. If the condition was a syntax error, the character position (column) within the statement unit is also given.

Some errors are general in nature and cannot be associated with a specific line number. General errors are indicated by the word NONE in the line number column.

Table B-1 explains the error codes used on the error summary. A set of three asterisks precedes each statement unit in the source listing on which the NDL processor performed a diagnostic function.

Occurrence of fatal errors listed in table B-1 does not necessarily terminate statement scanning, and additional errors in the same statement can be detected.

Occurrence of some errors does terminate statement scanning, so additional undetected errors beyond the one flagged can exist in any given statement. This prevents a cascade effect in the errors detected in case of the inadvertent omission of a parameter, keyword, or separator.

When a cascade of errors occurs for a single statement, and many of the error diagnostics issued for the statement do not seem to apply to it, the number of characters permitted in a statement unit has probably been exceeded. If the diagnosed statement runs to more than 72 meaningful characters (or if the preceding statement exceeds 72 meaningful characters) and is not divided into valid statement units, the NDL processor misinterprets the resulting character stream and can diagnose nonexistent errors.

Use of the keywords TIPTYPE, TC, and DT can cause partial or complete suspension of diagnostic processing for their respective LINE, GROUP, TERMINAL, DEVICE, or TERMDEV statements when a site-defined value is used as a parameter declaration. Occurrence of the following reserved words as valid declarations causes the NDL processor to suspend diagnostic checking of all parameters on the same statement that have values dependent on the site-defined parameter:

TT12 through TT14

TC28 through TC31

DT12

Parameters not dependent on the site-defined parameter receive normal diagnostic checking. For example:

A line with a TIPTYPE of MODE4 can have a terminal with a TC of TC30 and a DT of DT12, but still must have CA and TA values as part of its configuration.

A line with a TIPTYPE of TT12 can have a terminal with a TC of HASP as long as all parameters useable for terminals of any terminal class are correctly declared.

The NDL processor cannot perform all normal checks for completeness of definition in LINE, GROUP, TERMINAL, DEVICE, and TERMDEV statements if site-defined parameter values are present.

JOB DAYFILE MESSAGES

All of the informative and file access error messages produced in the dayfile for the job are listed in table B-2.

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES

Error Code	Severity	Message	Significance	Action
1	Fatal	DUPLICATE LABEL/ ELEMENT-NAME.	The element-name fields of at least two statements contain the character string on the flagged statement (check defname identifiers also).	Change one character in at least one field.
2	Fatal	INVALID STATEMENT NAME.	The statement name is not one of the reserved NDL statement-name words.	Check for a typographical error, then review section 2.
3	Fatal	INVALID STATEMENT DECLARATION.	A keyword is misspelled or misplaced, or a defname identifier cannot be expanded properly.	Check appendixes D and E.
4	Fatal	MISSING COLON AFTER ELEMENT NAME.	Self-explanatory.	Change and rerun the job.
5		Reserved for CDC use.		
6	Fatal	UNKNOWN KEYWORD OR DELIMITER.	A keyword is misspelled or a DEFINE statement is missing.	Change and rerun the job.
7	Fatal	EQUAL NOT ASSOCIATED WITH KEYWORD.	Misplaced or misused defname identifier, or typographical error.	Check the statement format in sections 2 through 8.
8	Fatal	PUNCTUATION ERROR.	An extra colon or other special character is present, or a required separator is missing.	Check the statement format in section 2.
9	Fatal	INVALID KEYWORD.	The keyword shown in the column to the left of the message cannot be used in this type of statement.	Check the statement format in sections 2 through 8.
10	Fatal	INVALID VALUE.	The value shown in the column to the left of the message cannot be used for the keyword you have associated it with.	Check the statement format in sections 2 through 8.
11	Fatal	NO PERIOD FOUND BEFORE ASTERISK.	The NDL processor cannot tell if the asterisk begins a comment or is an incorrect parameter declaration.	Insert a period or correct the declaration.
12	Fatal	NESTED DEFINE.	A DEFINE statement identifier cannot be used within another DEFINE statement.	Remove the identifier and rerun the job.
13		Reserved for CDC use.		
14	Fatal	STATEMENT OUT OF SEQUENCE.	The required hierarchy was violated.	Check figure 2-4.
15	Fatal	REQUIRED LABEL MISSING.	The element name field is empty when a name is needed.	Change and rerun the job.
16	Fatal	DEFINE-NAME CANNOT BE A RESERVED WORD.	The defname identifier you used is one of the reserved words disallowed in appendix D.	Change and rerun the job.
17	Warning	LABEL NOT REQUIRED FOR THIS STMT -- IGNORED.	The contents of the element name field are not needed.	None.
18	Fatal	ELEMENT-NAME IS TOO LONG.	The element name field contains a value more than seven characters long.	Change and rerun the job.

Error Code	Severity	Message	Significance	Action
19	Fatal	NO KEYWORD ASSOCIATED WITH VALUE.	Either an equals sign is missing, an extra separator is present, or the keyword was omitted.	Fix the statement and rerun the job.
20	Fatal	CONSECUTIVE COMMAS NOT ALLOWED.	Probably a typographical error; a parameter might be missing.	Delete the extra comma or insert the missing parameter.
21	Fatal	MISSING -END- STATEMENT.	The NDL processor cannot tell if your input file is complete.	Check that the file is undamaged; if so, add an END statement.
22	Fatal	FIRST CHARACTER AFTER PERIOD NOT AN ASTERISK.	The NDL processor cannot tell if you have misplaced sequence number field information or placed comments on the statement.	Insert an asterisk or reformat the statement and rerun the job.
23	Fatal	ELEMENT-NAME MUST BEGIN WITH A LETTER.	Self-explanatory.	Change and rerun the job.
24		Reserved for CDC use.		
25	Fatal	MISSING NFILE/LFILE STMT -- MUST BE 1ST IN DIV.	Each division of an NDL program begins at an NFILE or LFILE statement; at least one must be present.	Check statement placement or insert the required statement.
26		Reserved for CDC use.		
27	Fatal	NAME GENERATION SUPPRESSED, -NCIR- TOO LARGE.	The device names for switched circuits on an X.25 line are generated by adding the NCIR value to the element name root you specified; this cannot be done because the NCIR value is invalid.	Fix the NCIR value and rerun the job.
28	Fatal	GROUP NOT ALLOWED FOR X.25 LINES.	You cannot use a GROUP statement with this TIPTYPE value.	Replace the GROUP statement with a LINE statement; add LINE, TERMINAL, DEVICE, and TERMDEV statements if needed.
29	Fatal	KEYWORD NOT ALLOWED ON THIS STATEMENT.	Self-explanatory.	Check any TIPTYPE value associated with the statement.
30	Fatal	MISSING VALUE.	The keyword shown in the column to the left of this message is not a stand-alone keyword; either a value is missing or a separator is misplaced.	Add a value or delete the keyword and rerun the job.
31	Fatal	NAME GENERATION SUPPRESSED, ROOT NAME TOO LONG.	The element name of a line or device configured through a GROUP statement cannot be longer than five characters.	Shorten the name on the identified GROUP, DEVICE, or TERMDEV statement.
32	Fatal	NAME GENERATION SUPPRESSED, NO PORT FOR GROUP.	A valid port number is needed to use as the end of each element name for a line or device configured through a GROUP statement.	Add a valid PORT parameter to the GROUP statement and rerun the job.
33	Warning	PREVIOUSLY DEFINED TITLE OVER-WRITTEN.	You have more than one TITLE statement in the same division.	Check that an NFILE or LFILE statement is not missing.

Error Code	Severity	Message	Significance	Action
34	Warning	STORED ONLY FIRST 45 CHAR — IGNORED REST.	You cannot put more than 45 characters into the title field of a file summary listing; the TITLE statement contains more than 45.	None.
35	Warning	SUPERFLUOUS DATA AFTER END STMT, IGNORED.	The END statement might be misplaced; if so, your definition is incomplete.	Check statement placement; move the END statement if necessary.
36		Reserved for CDC use.		
37	Fatal	NAME GENERATION SUPPRESSED, NI TOO LARGE.	The GROUP statement is not valid; you are defining too many elements with one set of statements.	Change the NI value and add more statement groups if needed.
38	Fatal	NAME GENERATION SUPPRESSED, PORT TOO LARGE.	A valid port number is needed to use as the end of each element name for a line or device configured through a GROUP statement.	Change the PORT parameter on the GROUP statement and rerun the job.
39	Warning	DUPLICATE DECLARATION, OVERRIDES PREVIOUS ONE.	Self-explanatory; if this condition was not intended, you have defined the same characteristic twice.	Check the statement.
40	Warning	POSSIBLE MISSING STMT(S) PRECEDING THIS ONE.	The NDL processor expected to find other statements before this one.	Check statement placement.
41	Warning	CONSOLE EXPECTED HERE TO OWN PASSIVE DEVICES	Console definition is required as an owning console before a set of passive devices can be used.	Introduce a console device definition before the set of passive devices.
42 thru 99		Reserved for CDC use.		
100	Fatal	VALUE OUT OF RANGE.	The parameter shown in the column to the left of this message cannot have the value you specified.	Change and rerun the job.
101	Fatal	KEYWORD NOT ALLOWED WITH CTYP SPECIFIED.	Either the CTYPE value is incorrect, or you cannot declare this characteristic for a circuit of this type.	Change the statement and rerun the job.
102	Fatal	DUPLICATE PORT NUMBER.	Every line and trunk on an NPU must have a unique port number; the number specified appears in a previous LINE, TRUNK, or GROUP statement.	Change one of the port numbers to a correct value.
103	Fatal	REQUIRED PARAMETER MISSING.	Self-explanatory.	Refer to the statement format in sections 2 through 8.
104	Fatal	TIPTYPE NOT COMPATIBLE WITH LTYPE.	You have defined a line with physical characteristics that cannot be supported by the software servicing it; either the TIPTYPE value is wrong, or the LTYPE value is wrong.	Check for DEFINE statement identifier misplacement; change the incorrect value and rerun the job.
105	Fatal	LSPEED VALID FOR ASYNC NON-AUTO-REC LINES ONLY.	Self-explanatory; you cannot define a fixed speed for an automatic recognition line.	Remove the AUTO or XAUTO parameter, or the LSPEED parameter.

Error Code	Severity	Message	Significance	Action
106	Fatal	PARAMETER NOT ALLOWED WITH TIPTYPE SPECIFIED.	You cannot define an element using the chosen protocol as having the characteristic indicated by the keyword in the column to the left of this message.	Check that the TIPTYPE value is correct; if so, delete the invalid parameter.
107	Fatal	PARAMETER NOT VALID ON FIXED CONFIG LINES.	Self-explanatory.	The LINE or GROUP statement might be missing the AUTO or XAUTO parameter.
108	Fatal	PARAMETER VALID ON 2780 TERMINALS ONLY.	Either you have declared a characteristic this device cannot have, or the TC or STIP value is incorrect.	Change the incorrect TC or STIP value, or delete the invalid parameter.
109	Fatal	DUPLICATE CLUSTER ADDRESS.	The CA value must be unique for each terminal on a fixed configuration line; the value used appears in a previous TERMINAL or TERMDEV statement for the same line.	Change the parameter and rerun the job.
110	Fatal	PARAMETER NOT VALID WITH TC SPECIFIED.	Either the TC value is wrong, or you have declared a characteristic this device cannot have.	Change the incorrect TC value, or delete the invalid parameter indicated by the keyword in the column to the left of this message.
111	Fatal	VALUE NOT VALID WITH STIP SPECIFIED.	The keyword is permitted but the value you have used cannot be supported by the subcategory specified within the TIP.	Change the value and rerun the job.
112	Fatal	VALUE NOT VALID WITH TIPTYPE SPECIFIED.	The keyword is permitted but the value you have used cannot be supported by the software specified.	Change the value and rerun the job.
113	Fatal	-AUTOREC- NOT VALID ON FIXED CONFIG LINES.	Self-explanatory.	Check if the AUTO or XAUTO parameter is needed and missing; if not, delete the indicated parameter or specify a valid value.
114	Fatal	ALL PVC-S MUST BE SPECIFIED BEFORE ANY SVC-S.	A TERMINAL or TERMDEV statement might be misplaced; otherwise, the CTYP value might be wrong.	Change statement placement or the CTYP value, as appropriate.
115	Fatal	MAXIMUM NUMBER OF TERMINALS EXCEEDED FOR THE LINE.	The number of terminals allowed depends on the TIPTYPE and on whether the line is fixed configuration or automatic recognition.	Refer to sections 4 through 7; add LINE or GROUP statements as needed.
116	Fatal	MAX NUMBER OF DEVICES EXCEEDED FOR TERMINAL.	The number of devices allowed depends on the subTIPtype and on whether the line is fixed configuration or automatic recognition.	Refer to sections 4 through 7; delete the unsupportable TERMDEV or DEVICE statements.
117	Fatal	STIP OR TC MUST BE SPECIFIED.	You omitted both parameters; one or the other is needed.	Add either parameter.
118	Fatal	NEN VALUE MUST BE LESS THAN OR EQUAL TO NCIR.	You cannot enable more switched virtual circuits than you have defined; NCIR might be too small.	Change the incorrect parameter and rerun the job.

Error Code	Severity	Message	Significance	Action
119	Fatal	PARAMETER VALUE NOT A VALID NPU NAME.	The value you gave for the keyword in the column to the left of this message is not used as an element name in an NPU statement within the current network division.	Check the spelling of the name; perhaps an NPU statement is missing.
120	Fatal	DUPLICATE LOGICAL LINK DEFINITION.	You declared a previous LOGLINK statement with the same NCNAME value and host node number.	Correct the name or remove the extra LOGLINK statement.
121	Fatal	NCNAME NOT A VALID COUPLER OR NPU NAME.	You did not supply a COUPLER or NPU statement in this network division with this value as an element name.	Check the spelling; if correct, a statement is missing.
122		Reserved for CDC use.		
123	Fatal	MAXIMUM NUMBER OF COUPLERS EXCEEDED (2 PER NPU).	You have three or more COUPLER statements for the same NPU statement.	Delete the extra statement and rerun the job.
124	Fatal	DUPLICATE LOC VALUE.	The LOC value shown in the column to the left of this message appears in a previous COUPLER statement for this NPU; one of the values is wrong.	Change the incorrect statement and rerun the job.
125	Fatal	DUPLICATE NODE NUMBER.	Node numbers must be unique within the network; the number flagged has already been assigned to an element.	Change one of the node numbers and rerun the job.
126	Fatal	REQUIRED SUPLINK STATEMENT MISSING.	No supervision path exists for this NPU; one must be defined.	Add a SUPLINK statement and rerun the job.
127	Fatal	NO LOGICAL LINKS DEFINED TO THIS NPU.	A front-end NPU must have a logical link to its coupler; a remote NPU must have at least one link to a coupler in a front-end NPU.	Add a LOGLINK statement and rerun the job.
128	Fatal	LOGLINK (LLNAME) DOES NOT TERMINATE AT THIS NPU.	The logical link specified as the LLNAME value does not terminate in the same NPU as the supervisory link defined by this SUPLINK statement.	Check LLNAME value spelling; if correct, the LOGLINK statement NCNAME value is wrong. Fix the incorrect value.
129	Fatal	LLNAME VALUE NOT A VALID LOGICAL LINK NAME.	You specified an element name that is too long, or contains a character other than a letter or a digit, or does not begin with a letter.	Change the name and rerun the job.
130	Fatal	DUPLICATE TRUNK DEFINITION.	Either you specified the same port number for two trunks, or you have defined two trunks between one pair of NPUs.	Change a port number value or delete one TRUNK statement and rerun the job.
131	Fatal	PARAMETER NOT ALLOWED FOR 2741 TERMINALS.	Self-explanatory; either the STIP value or TC value is wrong, or the device cannot have the specified characteristic.	Change the STIP or TC value, or remove the parameter and rerun the job.
132	Fatal	AB, CN, B1, B2, CT, BS, EB, EL MUST ALL BE UNIQUE.	The parameter shown in the column to the left of the message has a value declared or used for another of these parameters.	Change one of the parameter values and rerun the job.
133	Fatal	PARAMETER NOT ALLOWED WITH DT SPECIFIED.	The parameter shown in the column to the left of the message has an inappropriate keyword, or the DT value is wrong.	Remove the parameter, change the keyword, or change the DT value.

Error Code	Severity	Message	Significance	Action
134	Fatal	PARAMETER NOT ALLOWED FOR M4A TERMINALS.	Either the STIP value is wrong, or the terminal cannot have the characteristic shown in the column to the left of the message.	Remove this parameter or change the STIP value.
135	Fatal	PARAMETER NOT ALLOWED WITH STIP/TC SPECIFIED.	Either the STIP value is wrong, or the TC value is wrong, or the terminal cannot have the characteristic shown in the column to the left of the message.	Remove this parameter or change the incorrect value.
136	Fatal	INCORRECT HN VALUE.	The host node number does not match with any host node number.	Check if HN is the same as the node number on a coupler statement.
137	Warning	ROUNDED TO NEXT RECOMMENDED VALUE (PARAM/VALUE).	The value you declared was not directly used. The value used and the parameter keyword are shown.	Refer to the text in section 6 or 7.
138	Fatal	VALUE NOT ALLOWED WITH DT SPECIFIED.	Either the value shown for the keyword is wrong, or the DT value is wrong.	Remove this parameter or change the incorrect value.
139	Fatal	-DO- VALUE IS NOT UNIQUE FOR DEVICE TYPE.	Device ordinals must be unique for all DEVICE statements with the DT value used within this terminal; one of the following is wrong: the DO value shown, the DO value on a previous DEVICE statement, or the DT value used.	Change the incorrect statement value and rerun the job.
140	Fatal	PL AND CP STREAM VALUES ARE NOT UNIQUE.	Two DEVICE statements with the indicated DT values have the same STREAM parameter value; either one of the DT values is wrong, or one of the STREAM values is wrong.	Change the incorrect parameter value and rerun the job.
141	Fatal	STREAM VALUE IS NOT UNIQUE FOR DEVICE TYPE.	Two DEVICE statements with the same DT value have the same STREAM parameter value; either one of the DT values is wrong, or one of the STREAM values is wrong.	Change the incorrect parameter value and rerun the job.
142	Fatal	MAXIMUM NUMBER OF DEVICES EXCEEDED.	The number of device statements allowed for a terminal or line depends on the protocol and whether the line is automatic recognition or fixed configuration.	Refer to the text of sections 4 through 7.
143	Fatal	-TA- VALUE IS NOT UNIQUE.	The TA value on this statement appears on another DEVICE statement specified for this terminal; one of the values is wrong.	Change the incorrect statement parameter value and rerun the job.
144	Fatal	CANNOT SPECIFY BOTH MFAM AND DFAM.	A default family name or a mandatory one can be specified, but not both.	Remove either parameter.
145	Fatal	VALUE FOR MUSER AND DUSER CANNOT BE ZERO.	Self-explanatory; no system user name default exists.	Remove the parameter or specify the value NONE.
146	Fatal	CANNOT SPECIFY BOTH MUSER AND DUSER.	A default user name or a mandatory one can be specified, but not both.	Remove either parameter.
147	Fatal	MAPPL OR PAPPL CANNOT BE A RESERVED APPL NAME.	You specified one of the programs disallowed in section 8.	Remove the parameter or change the value to a different name.

Error Code	Severity	Message	Significance	Action
148	Fatal	CANNOT SPECIFY BOTH MAPPL AND PAPPL.	A primary application or a mandatory one can be specified, but not both.	Remove either parameter.
149	Fatal	APPLICATION NAME SPECIFIED IS A RESERVED NAME.	You have tried to declare an APPL statement with an element name disallowed in section 8.	Remove the statement or change the element name.
150	Fatal	NCIR MUST BE LESS THAN OR EQUAL TO NSVC.	Self-explanatory; one of values is incorrect.	Change the incorrect value.
151	Fatal	BOTH ENDS OF TRUNK CONNECTED TO SAME NPU.	One of the NPU name values is wrong.	Change the incorrect parameter.
152	Fatal	MUTUALLY EXCLUSIVE PARAMETER(S) HAVE BEEN USED.	Dxxx and Xxxx parameters (named in the column to the left of the message) cannot appear in the same statement.	Review the statement description and delete inappropriate parameters.
153	Fatal	TOTAL LENGTH OF FAC EXCEEDS MAX (124 HEX DIGITS).	There can be any number of FAC parameters specified as long as the total number of hexadecimal digits does not exceed 124.	Decrease the number of FACs or the number of hexadecimal digits for one or more FACS.
154	Fatal	THIS LOGICAL LINK DOES NOT SPAN OVER ONE TRUNK.	No physical path exists for the link you defined; a TRUNK statement is missing or the NPU name is wrong.	Change the NPU name or add a TRUNK statement if the line actually exists; if not, delete this statement.
155	Fatal	NO CONSOLES DEFINED ON THIS TERMINAL.	The protocol for this line requires at least one console device per terminal.	Add a DEVICE statement or check for an incorrect DT value on an existing one.
156	Fatal	DTEA NOT ALLOWED WHEN DCE IS NOT SPECIFIED.	Either the DCE parameter is missing, a NO value for it should be YES, or the DTEA parameter is not needed.	Add a DCE=YES parameter or delete the DTEA parameter.
157	Fatal	ILLEGAL PORT ASSIGNMENT ON THIS NODE ID SHOWN.	One end of the trunk statement is defined to use a port number that is not sequentially assigned with other trunks on the NPU indicated in the column to the left of this message.	Check the NPU element name; if correct, check the PORT parameter values of all TRUNK statements; change one of the values.
158	Fatal	ILLEGAL VALUE AS A CONTROL CHARACTER.	The value is not allowed for the parameter indicated in the column to the left of this message. Some ASCII character code values are invalid for characters that affect terminal functions.	Check the range given for the parameter in sections 4 through 7; change the value to one within the range indicated in the text.
159	Warning	MAXIMUM NUMBER OF USER STATEMENTS EXCEEDED.	Your input file contains more than 1023 USER statements. The 1024th and subsequent statements are ignored.	Unneeded USER statements should be eliminated.
160	Fatal	STIP IS REQUIRED WHEN TC IS OF TYPE USER.	A CDC-written TIP cannot determine which subTIPtype is appropriate for the line because no terminal class characteristics were known when the TIP was written.	Add a STIP parameter with a value valid for the TIP.

Error Code	Severity	Message	Significance	Action
161	Warning	MISSING PARAMETERS FOR USER DEFINED TIPTYPE.	The CDC-written version of the network software requires values for the parameters listed in the column to the left of this message.	If the network software has not been modified to supply an internal default value, you should add the parameter to the statement.
162	Fatal	TSPEED INCOMPATIBLE FOR THIS AUTOREC INDICATOR.	Low TSPEED cannot be used with the XAUTO parameter, or high TSPEED cannot be used with the AUTO parameter.	Change the TSPEED or the AUTOREC indicator to be compatible.
163	Warning	VALUE OUT OF RANGE.	The value specified for the AL parameter is not within the range of 0 through 7.	Change the value specified for the AL parameter.
164 thru 999		Reserved for CDC use.		

TABLE B-2. NDL PROCESSOR DAYFILE MESSAGES

Message	Significance	Action
ABRT FROM xxxx - BAD NCF FILE RECORD.	Procedure NDLLIST located an NCF record with the wrong format. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	Regenerate NCF by running NDLP on the same source program. If the problem still exists, follow the site-defined procedure for software errors.
ABRT FROM xxxx - CANT READ LIN RECORDS.	Procedure NDLLIST located an error in the line record format during an I/O transfer of that record. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	Regenerate NCF by running NDLP on the same source program. If the problem still exists, follow the site-defined procedure for software errors.
ABRT FROM xxxx - CANT READ NCF RECDS.	Procedure NDLLIST located an error in an NCF record format during an I/O transfer of that record. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	Regenerate NCF by running NDLP on the same source program. If the problem still exists, follow the site-defined procedure for software errors.
ABRT FROM xxxx - FN VAL NOT DEVICE FN.	Procedure NDLLIST located a device parameter with an FN value outside the range allocated for device FNs. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	This indicates a design error. Follow the site-defined procedure for software errors.
ABRT FROM xxxx - FN VAL NOT LINE FN.	Procedure NDLLIST located a line parameter with an FN value outside the range allocated for line FNs. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	This indicates a design error. Follow the site-defined procedure for software errors.

Message	Significance	Action
ABRT FROM xxxx - FN VAL NOT TERM FN.	Procedure NDLLIST located a term parameter with an FN value outside the range allocated for term FNs. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	This indicates a design error. Follow the site-defined procedure for software errors.
ABRT FROM xxxx - INVALID RECORD TYPE.	Procedure NDLLIST located an invalid record type. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	Regenerate LCF/NCF by using NDLP. If the problem still exists, follow the site-defined procedure for software errors.
ABRT FROM xxxx - NO SUCH RECORD TYPE.	Procedure NDLLIST located an unrecognizable record type. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	Regenerate LCF/NCF by using NDLP. If the problem still exists, follow the site-defined procedure for software errors.
ABRT FROM xxxx - READ ERROR.	Procedure NDLLIST located a read error in NCF/LCF. The name of the procedure from which the ABORT message was generated is indicated by xxxx.	Regenerate LCF/NCF by using NDLP. If the problem still exists, follow the site-defined procedure for software errors.
ERROR IN LCF -- SUMMARY SUPPRESSED.	The NDL processor opened an existing local configuration file, but this file did not contain a valid file flag (see section 9). This message is also issued at the end of a file creation job when fatal errors were detected during processing.	The unverifiable file should be recreated or a different file used for NDL processor input.
ERROR IN NCF -- SUMMARY SUPPRESSED.	The NDL processor opened an existing network configuration file, but this file did not contain a valid file flag (see section 9). This message is also issued at the end of a file creation job when fatal errors were detected during processing.	The unverifiable file should be recreated or a different file used for NDL processor input.
INPUT FILE EMPTY.	The NDL processor attempted to open the file specified for job input but could not find any information.	Check job structure or verify contents of the input file. Rerun the job.
INSUFFICIENT FIELD LENGTH.	The NDL processor requires additional central memory to completely process all input statements that cause table generation. Excessive use of the DEFINE statement can cause the processor to need additional table space.	Remove as many NDL DEFINE statements as possible from the input file, or add an RFL statement to the command portion of the job and rerun the job.
INVALID CONTROL CARD OPTION.	The NDLP command used by the job contains a format or syntax error.	Correct the command and rerun the job.
NDLP COMPLETE.	The NDL processor has finished all possible work on its input and ended its execution.	Informative only.

Message	Significance	Action
lfn - NO ERRORS ENCOUNTERED.	A properly validated network definition file has been created or listed by the NDL processor and given the indicated local file name.	Informative only.
lfn - mmm ERRORS AND nn WARNINGS.	If mmm is not zero, the indicated number of fatal diagnostic message errors are described in the error summary listing produced by the NDL processor as part of the listing output file. A nonzero value for mmm indicates that any network definition file created by the job from the named division does not contain a valid file flag. If nn is not zero, the indicated number of nonfatal diagnostic message errors are described in the error summary listing. A nonzero value for nn does not affect the valid file flag of any network definition file created by the job.	Correct the NDL statements input and rerun the job if mmm is not zero.

This glossary defines terms unique to the description of the software presented in this manual. It does not contain terms defined in the American National Standard X3/TR-1 -- 77, American National Dictionary for Information Processing, unless those terms are used with a different meaning within this manual.

This glossary also contains terms whose interpretation within this manual is intended to be more constrained or different from that commonly made. Some terms used in other manuals for the network software are included for the reader's convenience when reconciling terminology.

Acknowledgment, Block –
A message returned to the sender confirming the delivery of one or more messages or blocks.

Application Connection Number (ACN) –
A number assigned by the Communications Supervisor program to identify a particular logical connection within an application program.

Application Interface Program (AIP) –
A group of routines that reside in the application program's field length to translate and buffer communication between the application program and the network.

Application Name (ANAME) –
Up to seven 6-bit display code letters or digits (the first must be a letter) used to identify an application program. It is used by another application program or by a terminal operator when connection to the application is requested.

Application Program –
A program resident in a host computer that uses the Network Access Method and provides an information storage, retrieval, and/or processing service to a remote user via the data communication network.

Archetype Terminal –
The specific terminal equipment possessing all of the device attributes used as defaults for the parameterization of one terminal class. Each terminal class has a corresponding archetype terminal.

Auto-Loading –
The process whereby a non-CDC network processing unit is loaded before joining the network. From the viewpoint of the Network Supervisor, the NPU loads itself automatically.

Automatic Login –
The process whereby one or more of the Network Validation Facility login dialog parameters is supplied to NVF from the local configuration file. Parameters supplied through automatic login configuration of a terminal suppress prompting for the corresponding dialog entries and can override any entries made from the terminal.

Automatic Recognition –
The process whereby the Terminal Interface Program identifies characteristics of a terminal when the terminal's communication line becomes active. The Terminal Interface Program determines these characteristics by various methods for lines configured for automatic recognition. The Communications Supervisor then matches these parameters against the descriptions of specific terminals in the network configuration file; the terminal with the closest match to the empirically determined parameters is automatically recognized as the terminal on the communication line.

Batch Device –
See Passive Device.

Block –
An arbitrary grouping of data bytes, transmitted as a single entity; blocks are formed and reformed by various software within the network, and by device hardware during input. See also Transmission Block, Downline Block, and Upline Block.

Block Limit –
The number of message blocks that can be awaiting delivery at any one time in either the host-to-NPU direction or the NPU-to-host direction for a single terminal or terminal device.

CDCNET –
See Control Data Distributed Communications Network.

Character –
Unless otherwise specified, references to characters in this manual are to CDC 6-bit display-coded characters.

Cluster –
Mode 4 devices grouped by a common cluster address.

Cluster Address –
The hardware address of a cluster. This term is used in several ways within mode 4 communications documentation, as shown in table C-1.

Communication Element –
Any entity that constitutes a point of input to, or output from, the data communication network. This includes terminal devices, communication lines, and application programs.

Communication Line –
A complete communication circuit between a terminal and its network processing unit.

Communications Control Program –
A portion of the network software that resides in a 255x series network processing unit. This software can include such routines as the Terminal Interface Program.

TABLE C-1. MODE 4 NOMENCLATURE EQUIVALENCE

Networks Nomenclature	Mode 4A Nomenclature	Mode 4C Nomenclature
Network processing unit	Data source	Control station
Cluster address	Site address	Station address
Cluster controller	Equipment controller	Station
Terminal address	Station address	Device address
Terminal	Equipment controller	Station
Device	Equipment	Device

Communications Supervisor -
A portion of the network software, written as an application program. The Communications Supervisor coordinates the network-oriented activities of its host computer and all of that host's communication elements.

Console -
See Interactive Device.

Constant Carrier -
A communication line with a transmission carrier signal that remains on continuously; failure is reported if the carrier signal received remains off for a period of time that equals or exceeds a failure verification period.

Contention Terminal -
When a terminal can input at the discretion of the terminal user and has an input rate that cannot be controlled directly; contrast with Controlled Terminal. Asynchronous terminals are contention terminals.

Control Data Distributed Communications Network (CDCNET) -

The collection of compatible hardware and software products offered by Control Data Corporation to interconnect computer resources into distributed communications networks.

Controlled Carrier -
A communication line with a transmission carrier signal that is raised and lowered with each block transmitted; failure is reported if the carrier signal received does not fluctuate in a similar fashion.

Controlled Terminal -
A terminal that places data on a communication line only in response to a poll. The maximum input rate for such a terminal can be controlled by controlling the polling rate. Mode 4 devices are controlled.

Coupler -
A hardware module that resides in a front-end network processing unit and links the network processing unit to a host computer.

Data -
Any portion of a message created by the source, exclusive of any information used to accomplish transmission of such a message.

Dedicated Line -
A communication line that is permanently connected between a terminal and a network processing unit. Contrast with Switched Line.

DEFINE -
An NDL statement that provides the macro-like capability of substituting an identifier in coding for a more complex entity. When the coding is processed, the identifier is interpreted as if it had been replaced by the complex entity. Also, a NOS command that creates permanent files.

Destination -
The device or application program designated to receive the message.

Destination Node -
The NPU node that directly interfaces to the destination of a data message block.

Device -
A portion or all of a terminal. This term is used in various ways within mode 4 communications documentation, as shown in table C-1.

Direct Access File -
In the context of NOS permanent files, a file that is accessed and modified directly.

Downline -
The direction of output flow, from host to device.

Downline Block -
A group of one or more logical lines (messages) transmitted from a host application program as a single unit.

Echoplex -
The process of returning received characters on a full-duplex line. Not all terminals on full-duplex communication lines are capable of echoplex operation.

Frame -
A block of data sent across a high-speed data link. The frame is the basic communication unit used in trunk (NPU-to-NPU) communications and provides high-data density in bit-serial format over data-grade lines, as well as data assurance.

Front-End NPU -
A network processing unit linked directly to a
host computer through a coupler.

Full Duplex -
Two-way simultaneous transmission on a communi-
cation line.

Half Duplex -
Two-way alternating transmission on a communi-
cation line.

Host -
A computer that executes application programs.

Host Availability Display (HAD) -
A terminal display that lists all the paths the
terminal user can currently use to gain access
to a host. The terminal user can control this
display by using the HD command. Using the HN
command sets or changes the path through which
the terminal is connected to the host. The HAD
is seen by the terminal user immediately after
entering HD or HD=Y, when first connecting to
an NPU, or when logging out from the host.

Host Node -
The node ID number of the NPU coupler that
directly interfaces with a host computer.

Host Operator (HOP)-
The administrative operator who manages the
communication elements of the network within
the local computer system by communicating with
the Communications Supervisor in the host
computer. Contrast with NPU operator. The
host operator is an administrative operator
within the network and is always the host
computer's operating system operator.

Input -
Information flowing upline from terminal to
host computer.

Interactive Device -
Any device capable of conducting both input and
output, making it capable of dialog with the
Network Validation Facility. Also known as a
console-type device. Contrast with passive
device.

Label -
The value declared as the element-name field of
an NDL statement.

Level -
For logical records, an octal number 0 through
17 in the system-supplied 48-bit marker that
terminates a short or zero-length PRU. In
input decks, an octal number specified on 7/8/9
or /*EOR cards.

Line -
A connection between an NPU and a terminal or a
group of terminals.

Local Computer System -
That portion of a network associated with one
copy of the Communications Supervisor and
controlled by a host operator.

Local Configuration File -
A file in the host computer system, containing
information on the access requirements and
restrictions of application programs or users
of that host. The file contains a list of the
application programs available for execution in
the host computer, and the devices that require
special login processing to access it. This is
a NOS direct access permanent file.

Local NPU -
An NPU that is connected to the host via a
coupler. Synonymous with Front-End NPU.
Contrast with Remote NPU.

Logical Connection -
A logical message path established between two
application programs or between a network
device and an application program. Until ter-
minated, the logical connection allows messages
to pass between the two entities.

Logical Line -
A single logical message transmitted between
two application programs or between a network
device and an application program. A logical
line can contain several physical lines.

Logical Link -
The portion of a logical connection defined by
host node and terminal node ID numbers.

Logical Record -
Under NOS, a data grouping that consists of one
or more PRUs terminated by a short PRU or zero-
length PRU. Equivalent to a system-logical-
record under NOS/BE.

Macromemory -
The portion of 255x Series network processing
unit memory that contains code involved in data
communication such as the Terminal Interface
Program.

Message -
A logical unit of information, as processed by
an application program; equivalent to a logical
line. When transmitted over a network, a mes-
sage can consist of one or more network data
blocks.

Micromemory -
The portion of 255x Series network processing
unit memory that contains code defining the
unit to itself.

Mode 4 -
A communication line transmission protocol that
requires the polling of sources for input to
the data communication network. Control Data
supports two types of mode 4 equipment, mode 4A
and mode 4C. Mode 4A equipment is polled
through the hardware address of the console
device, regardless of how many devices inter-
face to the network. Mode 4C equipment is
polled through separate hardware addresses,
depending on the point each device uses to
interface with the network.

Neighbor Node -
The node ID number associated with an NPU at one end of a trunk by the NPU at the other end of the trunk. The neighbor node can be the same as a terminal node.

Network Access Method (NAM) -
A software package that provides a generalized method of using a communication network for switching, buffering, queuing, and transmitting data.

Network Configuration File -
A network definition file in the host computer, containing information on the network elements and permissible linkages between them. The status of the elements described in this file is modified by the host or NPU operator in the course of managing the network through the Communications Supervisor. This is a NOS direct access permanent file.

Network Definition File -
Either of the two types of NDL program output files that determine the configuration of the network. This can be a network configuration file or a local configuration file.

Network Definition Language (NDL) -
The compiler-level language used to define the network configuration file and local configuration file contents.

Network Definition Language Processor (NDL Processor) -
The network software module that processes an NDL program as an off-line batch job to create the network definition files and other NDL program output.

Network Element -
Any configurable entity supervised by the Communications Supervisor or loaded by the Network Supervisor. A network element consists of any entity in the total computer and terminal network; this term is often applied to the data communication network entities comprising the NPUs and their linkages.

Network Processing Unit (NPU) -
The collection of hardware and software that switches, buffers, and transmits data between terminals and host computers.

Network Supervisor -
A portion of the network software, written as a NAM application program. The Network Supervisor loads all NPUs in the communication network that cannot load themselves.

Node -
A network element that creates, absorbs, switches, and/or buffers message blocks.

Normalized Mode -
The initial transmission mode of all console devices in the network. In normalized mode, the Terminal Interface Program performs limited line editing functions on input, converts all terminal character codes to 7-bit ASCII codes, and performs upline blocking based on input of end-of-line and end-of-block codes. Normalized mode output is treated in the opposite manner; ASCII codes received from the host are con-
verted to terminal character codes and codes to reposition and control the output mechanism and are inserted at appropriate places in the output data.

NPU Load File -
The permanent file within the host computer system that contains variants of the CCP software appropriate for loading into the network's 255x Series NPUs; created by the NPU load file generator and used by the Network Supervisor.

NPU Operator (NOP)-
The administrative operator who manages the hardware, linkages, and other network elements of the data communication network by communicating with the Communications Supervisor in a host computer. Contrast with host operator. The host operator can also be an NPU operator, but the NPU operator need not be the operating system operator for the host computer.

Output -
Information flowing downline from host to device.

Owning Console -
The interactive device associated with a given passive device. The owning console enters commands that control the operation of the passive device and is always a device within the same terminal.

Packet-Switching Network (PSN) -
A network that provides data communications service between various terminals and computer systems or networks.

Terminal interface to a PSN is defined by the packet assembly/disassembly (PAD) access. PSN interface with a NOS network is defined by the X.25 protocol. Packet-switching networks are sometimes called public data networks (PDNs).

PAD -
See X.25 Packet Assembler/Disassembler.

PAD SubTIP -
A subTIP of the X.25 TIP that allows asynchronous ASCII terminals to communicate over a packet-switching network.

Parameterization -
The process whereby all of the configurable characteristics of a specific model of terminal are reconciled with the characteristics of that terminal's general terminal class. All characteristics not specifically declared for a given device are inferred from the terminal's assigned terminal class. Characteristics can be declared through the device definition in the network configuration file, or by the terminal user through dialog with the Terminal Interface Program, or by an application program servicing the device.

Passive Device -
Any device incapable of conducting both input and output and therefore incapable of dialog with the Network Validation Facility. Batch unit record peripherals are typical examples of passive devices. Also known as a nonconsole device. Contrast with Interactive Device.

Peripheral Processor Unit (PPU) —
The hardware unit within the host computer that performs physical input and output through the computer's data channels.

Physical Line —
A string of data that is determined by the physical characteristics of the terminal (page width or line feed). Contrast with Logical Line, which is determined by a carriage return or other forwarding signal.

Physical Record Unit (PRU) —
Under NOS, the amount of information transmitted by a single physical operation of a specified device. The size of a PRU depends on the device:

Device	Size in Number of 60-Bit Words
Mass storage	64
Tape in SI format with binary data	512
Tape in I format	512
Tape in other format	Undefined

A PRU that is not full of user data is called a short PRU; a PRU that has a level terminator but no user data is called a zero-length PRU.

Port Number —
The hexadecimal number assigned to the communication line ports of an NPU; physically identified by a thumbwheel on a communications line adapter of the NPU.

Priority —
The condition when traffic through the network is maintained preferentially for one or more devices out of all devices producing network traffic. Devices with priority are the last devices for which network traffic is suspended when traffic must be temporarily stopped because the network is operating at capacity. Devices with priority do not otherwise receive any preferential treatment of their input or output.

Program Initiation Control Block (PICB) —
The identifier of a load module for the CCP software contained in the NPU load file; the PICB determines what variant of CCP is downline loaded into a given NPU by the Network Supervisor.

PRU Device —
Under NOS, a mass storage device or a tape in SI or I format, so called because records on these devices are written in PRUs.

Random File —
In the context of the NOS operating system, a file with the random bit set in the file environment table in which individual records are accessed by their relative PRU numbers.

Remote NPU —
A network processing unit linked indirectly to a host computer through another network processing unit.

Sequential —
A file organization in which records are stored in the order in which they are generated.

Short PRU —
A PRU that does not contain as much user data as the PRU can hold and that is terminated by a system terminator with a level number. Under NOS, a short PRU defines EOR.

Site —
All of the hosts and NPUs in a network under single administrative control. All of the NPUs belonging to a site must run the same release level of CCP. The NCFs used by the site must be built by the same level of NDLP as the release level of CCP.

Source —
The device or host computer program that created the message.

Source Node —
The node that interfaces directly to the source of a data message block.

Station —
A provider and/or recipient of data messages; usually synonymous with a terminal or a grouping of devices. This term is used in various ways within mode 4 communication documentation, as shown in table C-1.

Switched Line —
A communication line connected with one network processing unit but able to be connected to any one of several terminals via a switching mechanism, such as a dialed telephone line.

Terminal —
An entity, external to the communications network but connected to it via a communication line, that supplies input messages to, and/or accepts output messages from, an application program. A terminal can comprise only one device (interactive terminals) or several devices (batch terminals).

Terminal Address —
The hardware address of a mode 4 station. This term is used in various ways within mode 4 communication documentation, as shown in table C-1.

Terminal Class —
An NDL parameter describing the physical attributes of a group of similar terminals, in terms of an archetype terminal for the group.

Terminal Interface Program (TIP) —
A portion of the Communications Control Program that provides an interface for terminals connected to a 255x Series network processing unit. The TIP performs character conversion to and from 7-bit ASCII, limited editing of the input and output stream, parity checking, and so forth.

Terminal Name -
   A name of up to seven letters and digits known
   to the network and used to identify a device to
   the host or NPU operator.

Terminal Node -
   The node ID number associated with an NPU that
   interfaces with a terminal.

Terminal User -
   The person operating the controls of a device.

Transmission Block -
   During input, a group of one or more logical
   lines (messages of one or more network data
   blocks) transmitted as a single unit; during
   output, a group of one or more physical lines
   transmitted as a single unit.

Transparent Mode -
   A software feature provided by the Network
   Access Method and the network processing unit
   TIP. When transparent mode transmission occurs
   between an application program and a device,
   the Network Access Method does not convert data
   to or from display code, and the TIP does not
   completely edit the character stream to insert
   or remove terminal protocol control codes or
   convert the characters to or from 7-bit ASCII
   code. When no parity is in effect for the
   device and transparent mode transmission
   occurs, all 8 bits of the character byte can be
   used to represent characters in 256-character
   sets (such as EBCDIC).

Trunk -
   The communication line connecting two network
   processing units.

Upline -
   The direction of input flow from terminal to
   host computer.

Upline Block -
   A group of data bytes representing a single
   physical line of input; a logical line (a
   single message) can consist of more than one
   upline network data block.

User Name -
   The NOS validation file parameter entered by
   the terminal user during the Network Validation
   Facility login procedure.

X.25 Packet Assembler/Disassembler (PAD) -
   The access mechanism used by an unintelligent
   asynchronous terminal to access a packet-
   switching network; the PAD is usually supplied
   by the packet-switching network. The PAD is
   defined by the CCITT Standard X.3, and the
   interfaces between the PAD and the terminal and
   the PAD and the NPU are defined in CCITT Stan-
   dards X.29 and X.28.

Zero-Length PRU -
   A PRU that contains system information but no
   user data. Under NOS, a zero-length PRU de-
   fines EOF.

The Network Definition Language contains two types of reserved words: statement names and keywords that control statement interpretation, and reserved words that define values. This appendix lists reserved words in both categories; words reserved for future implementation are also listed.

RESERVED STATEMENT
NAMES AND KEYWORDS

The NDL Processor interprets the following words as statement identifiers or as keywords within statements. You cannot use these words as DEFINE statement identifiers and should not use them as element names or as values for user-defined parameters:

<div style="display: flex;">
<div>

AB
ABL
ACCLEV
AL
ANAME
APPL
ARSPEED
AUTO
AUTOCON
BCF
BR
BS
BSC
B1
B2
CA
CI
CN
CO
COLLECT
COMMENT
COUPLER
CP
CSET
CT
CTYP
DBL
DBZ
DCE
DEFINE
DEVICE
DFAM
DFL
DHOST
DI
DLC
DLTO
DLX
DMP
DNODE
DO
DOMAIN
DPLR
DPLS
DT
DTE

</div>
<div>

DTEA
DUSER
EBO
EBR
EBX
ELO
ELR
ELX
END
EOF
EP
FAC1 thru FAC31
FAM
FASTSEL
FRAME
GROUP
HD
HN
HNAME
IC
IMDISC
IN
INCALL
KDSP
LCN
LFILE
LI
LINE
LK
LLNAME
LOC
LOGLINK
LSPEED
LTYPE
MAPPL
MC
MCI
MFAM
MLI
MREC
MUSER
NAME1
NAME2
NCIR
NCNAME
NEN
NETOSD
NFILE
NI
NODE
NOLOAD
NOLOAD1
NOLOAD2
NPU
NSVC
N1
N2
OC
OP
OPGO
OUTCALL
PA
PAD
PAPPL
PFAM

</div>
</div>

PG
PID
PL
PORT
PR
PRI
PRID
PRIV
PRU
PSN
PT
PUSER
PW
P1
P2
P90 thru P99
RC
RCOUNT
RIC
RTIME
RTS
SDT
SERVICE
SHOST
SNODE
STIP
STREAM
SUPLINK
TA
TC
TERMDEV
TERMINAL
TIPTYPE
TITLE
TMC
TMTO
TMX
TMY
TRUNK
TSPEED
UBL
UBZ
UDATA
UID
UNAME
USER
VARIANT
W
WS
XAUTO
XBZ
XLC
XLTO
XLX
XLY

RESERVED VALUE MNEMONIC WORDS

The NDL processor recognizes the following words as predefined values in parameter declarations. You should not use these words as DEFINE statement identifiers or as element names:

AP
APLBP
APLTP
ASCII
ASYNC
AUTOREC
A1
A2
A6

A9
BCD
BK
BSC
B6
CCP
CDSN
CON
CORAPL
CORRES
CP
CR
C120
DATAPAC
DI
DT12
E
EB
EBCD
EBCDAPL
EBCDIC
EL
HASP
HPOST
HPRE
H1
H2
H2000
KB
LP
MODE4
M33
M4A
M4C
M40
N
NO
NONE
N2741
O
PAD
PL
POST
PR
PRE
PRIMARY
PSN253 thru PSN255
PT
PVC
SDT12
SDT13
SDT14
SDT15
SECOND
SVC
S1
S2
S3
S4
TC28 thru TC31
TELENET
TRNSPAC
TT12 thru TT14
TYMNET
T4014
UNINET
USER
XAA
X25
X364
YES
Z
6BIT
8BIT
26

29
110
134
150
200UT
300
600
711
713
714
714X

721
734
752
1200
2400
2741
2780
3270
3780
4800
9600

FORMAL SYNTAX

The syntax of the Network Definition Language is formally described in Backus-Naur Form, as shown in figure E-1. This formal syntax is provided as background information only; the syntax used in coding an NDL program is described in sections 2 through 8.

```
The following list defines the terms of statement definition:

    <asterisk>   :  :=  *
    <asterisks>  :  :=  <asterisk>  |  <asterisks> <asterisk>
    <null>   :  :=  <>
    <blank>  :  :=  <   >
    <blanks>   :  :=  <blank>  |  <blanks> <blank>
    <nblanks>  :  :=  <blanks>  |  <null>
    <letter>   :  :=  A | B | ... | Z
    <digit>  :  :=  0 | 1 | ... | 9 | A | ... | F
    <alpha>  :  :=  <letter>  |  <digit>
    <ID>   :  :=  <letter>  |  <ID> <digit>
    <number>  :  :=  <digit>  |  <number> <digit>
    <equals>  :  :=  =
    <comma>   :  :=  <nblanks> , <nblanks>
    <period>  :  :=  <nblanks> . <nblanks>
    <colon>   :  :=  :
    <separator>  :  :=  <comma>  |  <blanks>
    <username>  :  :=  <ID>  |  <ID> <asterisks>  |  <asterisks> <ID>  |  <username> <asterisks>  |
                           <asterisks> <username>
    <value>  :  :=  <number>  |  <ID>  |  <username>


The following list defines the content of statement definitions:

    <element-name>  :  :=  <ID>  |  <null>
    <statement-ID>  :  :=  <ID>
    <keyword>  :  :=  <ID>
    <keywords>  :  :=  <keyword>  |  <keywords> <keyword>
    <declaration>  :  :=  <keywords>  |  <keyword> <equals> <value>
    <declarations>  :  :=  <declaration>  |  <declarations> <separator> <declaration>
    <statement-end>  :  :=  <period>


The following defines the general form of an NDL statement:

    <NDL-statement>  :  :=  <element-name> <colon> <nblanks> <statement-ID> <separator>
                              <declarations> <statement-end>
```

Figure E-1. Formal Syntax of NDL in Backus-Naur Form

STATEMENT SUMMARIES

This appendix summarizes the formats of all recognized NDL statements. Formats are presented in a manner similar to that used in sections 2 through 8, where the valid value declarations are described in detail. The following formats are listed alphabetically by statement name; default values are shown; they are underlined when no dependencies for the default value exist.

▌ APPL Statement

service: APPL [, PRIV [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , UID [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , DI [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , KDSP [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] ,

PRU [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , NETXFR [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , RS [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , MXCOPYS=mxcopys] .

▌ COMMENT Statement

$\left\{\begin{matrix}\text{COMMENT}\\ *\end{matrix}\right\}$ [, string] [.]

▌ COUPLER Statement

coupler: COUPLER, NODE=node [, HNAME=hostnam, LOC= $\left\{\begin{matrix}\underline{\text{PRIMARY}}\\ \text{SECOND}\end{matrix}\right\}$] .

▌ DEFINE Statement

defname: DEFINE, value declaration [, value declaration,] ... , [value declaration] .

▌ DEVICE Statement

device: DEVICE [, DT=devtyp, SDT=subdt, TA= $\left\{\begin{matrix}\text{AUTOREC}\\ \text{trmaddr}\end{matrix}\right\}$, ABL=abl, DBZ=dwnlsiz, UBZ= $\left\{\begin{matrix}\underline{100}\\ \text{upbsize}\end{matrix}\right\}$,

DBL=dwnblim, UBL=upblim, XBZ=xmitsiz, DO= $\left\{\begin{matrix}\underline{1}\\ \text{devord}\end{matrix}\right\}$, STREAM= $\left\{\begin{matrix}\text{AUTOREC}\\ \text{streamno}\end{matrix}\right\}$,

AUTOCON [= $\left\{\begin{matrix}\text{YES}\\ \text{NO}\end{matrix}\right\}$] , PRI [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , DI [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , HN=node, HD [= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}$] , LK= $\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\\ \text{CCP}\end{matrix}\right\}$,

AB= $\left\{\begin{matrix}\text{CCP}\\ \text{ab}\end{matrix}\right\}$, BR= $\left\{\begin{matrix}\text{YES}\\ \text{CCP}\\ \underline{\text{NO}}\end{matrix}\right\}$, CP= $\left\{\begin{matrix}\text{YES}\\ \text{CCP}\\ \text{NO}\end{matrix}\right\}$, BS= $\left\{\begin{matrix}\text{CCP}\\ \text{bs}\end{matrix}\right\}$, B1= $\left\{\begin{matrix}\text{CCP}\\ \text{b1}\end{matrix}\right\}$, B2= $\left\{\begin{matrix}\text{CCP}\\ \text{b2}\end{matrix}\right\}$, CI= $\left\{\begin{matrix}\text{CCP}\\ \text{ci}\end{matrix}\right\}$,

CN= $\left\{\begin{matrix}\text{CCP}\\ \text{cn}\end{matrix}\right\}$, CT= $\left\{\begin{matrix}\text{CCP}\\ \text{ct}\end{matrix}\right\}$, DLC= $\left\{\begin{matrix}\text{CCP}\\ \text{dlc}\end{matrix}\right\}$, DLTO= $\left\{\begin{matrix}\text{YES}\\ \text{CCP}\\ \underline{\text{NO}}\end{matrix}\right\}$, DLX= $\left\{\begin{matrix}\text{CCP}\\ \text{dlx}\end{matrix}\right\}$, EBX= $\left\{\begin{matrix}\text{CCP}\\ \text{ebx}\end{matrix}\right\}$, EBR= $\left\{\begin{matrix}\text{CR}\\ \text{LF}\\ \text{CCP}\\ \text{CL}\\ \text{NO}\end{matrix}\right\}$,

(continued on next page)

$$EBO= \begin{Bmatrix} CCP \\ EL \end{Bmatrix} , \quad ELX= \begin{Bmatrix} OD \\ CCP \\ elx \end{Bmatrix} , \quad ELR= \begin{Bmatrix} CR \\ LF \\ CCP \\ CL \\ NO \end{Bmatrix} , \quad ELO= \begin{Bmatrix} EB \\ CCP \end{Bmatrix} , \quad EP= \begin{Bmatrix} YES \\ CCP \\ \underline{NO} \end{Bmatrix} , \quad XLC= \begin{Bmatrix} CCP \\ xlc \end{Bmatrix} , \quad XLTO= \begin{Bmatrix} YES \\ CCP \\ \underline{NO} \end{Bmatrix} ,$$

$$XLX = \begin{Bmatrix} CCP \\ xlx \end{Bmatrix} , \quad XLY= \begin{Bmatrix} CCP \\ xly \end{Bmatrix} , \quad IC= \begin{Bmatrix} YES \\ CCP \\ \underline{NO} \end{Bmatrix} , \quad IN= \begin{Bmatrix} BK \\ CCP \\ KB \\ \underline{PT} \end{Bmatrix} , \quad LI= \begin{Bmatrix} CCP \\ li \end{Bmatrix} , \quad OP= \begin{Bmatrix} PR \\ DI \\ CCP \\ PT \end{Bmatrix} , \quad OC= \begin{Bmatrix} YES \\ CCP \\ \underline{NO} \end{Bmatrix} ,$$

$$PA= \begin{Bmatrix} E \\ O \\ CCP \\ N \\ Z \\ I \end{Bmatrix} , \quad PG= \begin{Bmatrix} YES \\ CCP \\ \underline{NO} \end{Bmatrix} , \quad PL= \begin{Bmatrix} CCP \\ pl \end{Bmatrix} , \quad PW= \begin{Bmatrix} CCP \\ pw \end{Bmatrix} , \quad RTS\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] , \quad MCI= \{mc\} , \quad MLI= \{ml\} ,$$

P90=fv90, ... , P99=fv99] .

END Statement

END.

GROUP Statement

group: GROUP, PORT=port, LTYPE=ltype [, TIPTYPE=tiptype,

$$AUTO\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] , \quad XAUTO\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] , \quad DI\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] , \quad LSPEED= \begin{Bmatrix} 300 \\ lspeed \end{Bmatrix} ,$$

$$AL=acclev, \quad ARSPEED=arspeed, \quad IMDISC\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] , \quad RC\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] ,$$

$$P90=fv90, \ldots , P99=fv99, \quad NI= \begin{Bmatrix} \underline{1} \\ iter \end{Bmatrix}] .$$

INCALL Statement

INCALL, FAM=famname, UNAME=usernam, ANAME=appname [, SNODE=srcnode,

$$DNODE=dstnode, \quad PRI\ [= \begin{Bmatrix} YES \\ \underline{NO} \end{Bmatrix}] , \quad DBL=dwnblim, \quad ABL=abl, \quad DBZ=dwnlsiz,$$

WS=sndsiz, WR=recsiz, DPLS=dpls, DPLR=dplr, SHOST=srchost, UBL=upblim, UBZ=upbsize,

PORT=portnum, DTEA=dtea, COLLECT, FASTSEL, FAC1=faccode, ... , FAC31=faccode] .

LFILE Statement

lfile: LFILE.

LINE Statement

```
line:  LINE, PORT=port, LTYPE=ltype [ , TIPTYPE=tiptype, AUTO [= {YES / NO} ] , XAUTO [= {YES / NO} ] ,

       DI [= {YES / NO} ] , LSPEED= {300 / lspeed} , AL=acclev, DFL=dfl, FRAME=frame, RTIME=timer,

       RCOUNT=count, NSVC=svcirc, PSN=psn, DCE [= {YES / NO} ] , DTEA=locadr, ARSPEED [= {YES / NO} ] ,

       IMDISC [= {YES / NO} ] , LCN=lcn, RC [= {YES / NO} ] , P90=fv90, ... , P99=fv99 ] .
```

LOGLINK Statement

```
loglink:  LOGLINK, NCNAME=ncname [, DI [= {YES / NO} ] ] .
```

NFILE Statement

```
nfile:  NFILE.
```

NPU Statement

```
npu:  NPU, NODE=node, VARIANT=picbname [, OPGO [= {YES / NO} ] , DMP [= {YES / NO} ] ] .
```

OUTCALL Statement

```
OUTCALL, NAME1=name1, NAME2/PID=name2/pidname [, NETOSD=path, SNODE=srcnode, DNODE=dstnode,

        ACCLEV=acclev, PRI [= {YES / NO} ] , DBL=dwnblim, ABL=abl, DBZ=dwnlsiz, DHOST=dsthost,

        SHOST=srchost, PORT=portnum, WS=windsiz, DPLS=dpls, DTEA=dtea, UBL=upblim, UBZ=upbsize,

        FAC1=faccode, ... , FAC31=faccode, PRID=protid, SERVICE=service, DOMAIN=domain,

        UDATA=udata ] .
```

SUPLINK Statement

```
SUPLINK, LLNAME=loglink.
```

TERMDEV Statement ▮

$$\text{device: TERMDEV } \left[\text{, STIP} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{stiptyp}\end{array}\right\}, \text{ TC=trmclas, CSET} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{charset}\end{array}\right\}, \text{ TSPEED} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{trmsped}\end{array}\right\},\right.$$

$$\text{CA} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{clsadr}\end{array}\right\}, \text{ RIC} = \left\{\begin{array}{l}\text{YES}\\\text{NO}\end{array}\right\}, \text{ CO} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{conord}\end{array}\right\}, \text{ BCF } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right], \text{ MREC=mrec, W=pacwndw,}$$

$$\text{NCIR=numcirc, PAD=string, NEN=encirc, COLLECT } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right], \text{ DT=devtyp, SDT=subdt,}$$

$$\text{TA} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{trmadr}\end{array}\right\}, \text{ ABL=abl, DBZ=dwnlsiz, UBZ} = \left\{\begin{array}{l}\text{100}\\\text{uplsiz}\end{array}\right\}, \text{ DBL=dwnblim, UBL=upblim, XBZ=xmitsiz,}$$

$$\text{DO} = \left\{\begin{array}{l}\text{1}\\\text{devord}\end{array}\right\}, \text{ STREAM=streamno, AUTOCON } \left[= \left\{\begin{array}{l}\text{YES}\\\text{NO}\end{array}\right\}\right], \text{ PRI } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right],$$

$$\text{DI } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right], \text{ HN=node, HD } \left[= \left\{\begin{array}{l}\text{YES}\\\text{NO}\end{array}\right\}\right], \text{ LK} = \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\\\text{CCP}\end{array}\right\}, \text{ AB} = \left\{\begin{array}{l}\text{CCP}\\\text{ab}\end{array}\right\}, \text{ BR} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\}, \text{ CP} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\text{NO}\end{array}\right\},$$

$$\text{BS} = \left\{\begin{array}{l}\text{CCP}\\\text{bs}\end{array}\right\}, \text{ B1} = \left\{\begin{array}{l}\text{CCP}\\\text{b1}\end{array}\right\}, \text{ B2} = \left\{\begin{array}{l}\text{CCP}\\\text{b2}\end{array}\right\}, \text{ CI} = \left\{\begin{array}{l}\text{CCP}\\\text{ci}\end{array}\right\}, \text{ CN} = \left\{\begin{array}{l}\text{CCP}\\\text{cn}\end{array}\right\}, \text{ CT} = \left\{\begin{array}{l}\text{CCP}\\\text{ct}\end{array}\right\}, \text{ DLC} = \left\{\begin{array}{l}\text{CCP}\\\text{dlc}\end{array}\right\},$$

$$\text{DLTO} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\}, \text{ DLX} = \left\{\begin{array}{l}\text{CCP}\\\text{dlx}\end{array}\right\}, \text{ EBX} = \left\{\begin{array}{l}\text{CCP}\\\text{ebx}\end{array}\right\}, \text{ EBR} = \left\{\begin{array}{l}\text{CR}\\\text{LF}\\\text{CCP}\\\text{CL}\\\text{NO}\end{array}\right\}, \text{ EBO} = \left\{\begin{array}{l}\text{CCP}\\\text{EL}\end{array}\right\}, \text{ ELX} = \left\{\begin{array}{l}\text{OD}\\\text{CCP}\\\text{elx}\end{array}\right\}, \text{ ELR} = \left\{\begin{array}{l}\text{CR}\\\text{LF}\\\text{CCP}\\\text{CL}\\\text{NO}\end{array}\right\},$$

$$\text{ELO} = \left\{\begin{array}{l}\text{EB}\\\text{CCP}\end{array}\right\}, \text{ EP} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\}, \text{ XLC} = \left\{\begin{array}{l}\text{CCP}\\\text{xlc}\end{array}\right\}, \text{ XLTO} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\}, \text{ XLX} = \left\{\begin{array}{l}\text{CCP}\\\text{xlx}\end{array}\right\}, \text{ XLY} = \left\{\begin{array}{l}\text{CCP}\\\text{xly}\end{array}\right\}, \text{ IC} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\},$$

$$\text{IN} = \left\{\begin{array}{l}\text{BK}\\\text{KB}\\\underline{\text{CCP}}\\\text{PT}\end{array}\right\}, \text{ LI} = \left\{\begin{array}{l}\text{CCP}\\\text{li}\end{array}\right\}, \text{ OP} = \left\{\begin{array}{l}\text{PR}\\\text{DI}\\\text{CCP}\\\text{PT}\end{array}\right\}, \text{ OC} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\}, \text{ PA} = \left\{\begin{array}{l}\text{E}\\\text{O}\\\text{CCP}\\\text{N}\\\text{Z}\\\text{I}\end{array}\right\}, \text{ PG} = \left\{\begin{array}{l}\text{YES}\\\text{CCP}\\\underline{\text{NO}}\end{array}\right\}, \text{ PL} = \left\{\begin{array}{l}\text{CCP}\\\text{pl}\end{array}\right\},$$

$$\text{PW} = \left\{\begin{array}{l}\text{CCP}\\\text{pw}\end{array}\right\}, \text{ EOF} = \left\{\begin{array}{l}\text{YES}\\\text{NO}\end{array}\right\}, \text{ RTS } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right], \text{ MCI = \{mc\} , MLI = \{ml\} , P90=fv90, ... , P99=fv99 }\right].$$

TERMINAL Statement ▮

$$\text{TERMINAL } \left[\text{, STIP} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{stiptyp}\end{array}\right\}, \text{ TC=trmclas, CSET} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{charset}\end{array}\right\}, \text{ TSPEED} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{trmsped}\end{array}\right\}, \text{ CA} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{clsadr}\end{array}\right\},\right.$$

$$\text{RIC} = \left\{\begin{array}{l}\text{YES}\\\text{NO}\end{array}\right\}, \text{ CO} = \left\{\begin{array}{l}\text{AUTOREC}\\\text{conord}\end{array}\right\}, \text{ BCF } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right], \text{ MREC=mrec, W=pacwndw, CTYP=cirtyp,}$$

$$\text{NCIR=numcirc, PAD=string, EOF} = \left\{\begin{array}{l}\text{YES}\\\text{NO}\end{array}\right\}, \text{ NEN=encirc, COLLECT } \left[= \left\{\begin{array}{l}\text{YES}\\\underline{\text{NO}}\end{array}\right\}\right]\left.\right].$$

TITLE Statement ▮

$$\text{TITLE } [\text{, string }].$$

TRUNK Statement

trunk: TRUNK, N1=npu1, N2=npu2, P1=port1, P2=port2,   3-7

[NOLOAD1 [= $\left\{\begin{matrix} YES \\ \underline{NO} \end{matrix}\right\}$], NOLOAD2 [= $\left\{\begin{matrix} YES \\ \underline{NO} \end{matrix}\right\}$], FRAME=frame, DI [= $\left\{\begin{matrix} YES \\ \underline{NO} \end{matrix}\right\}$]] .

USER Statement

device: USER [, MFAM= $\left\{\begin{matrix} 0 \\ manafam \\ \underline{NONE} \end{matrix}\right\}$, MUSER= $\left\{\begin{matrix} manausr \\ \underline{NONE} \end{matrix}\right\}$, MAPPL= $\left\{\begin{matrix} manappl \\ \underline{NONE} \end{matrix}\right\}$, DFAM= $\left\{\begin{matrix} 0 \\ deftfam \\ \underline{NONE} \end{matrix}\right\}$,   8-2

DUSER= $\left\{\begin{matrix} deftusr \\ \underline{NONE} \end{matrix}\right\}$, PFAM=pfam, PUSER=puser, PAPPL= $\left\{\begin{matrix} primapp \\ \underline{NONE} \end{matrix}\right\}$] .

The NDL processor allows you to configure more network elements than the released version of the CDC network software supports. There are no maximum allowed numbers of most NDL statements. The Network Definition Language can therefore be used to define configurations supported by site-written variants of the released network software.

The released version of the CDC network software requires upper limits on the number of elements configured for each host. These limits avoid excessive central memory usage by the network software. The limits consist of a maximum number of elements, as described in table F-1.

Other constraints are imposed by the types of elements in the network and by the method used to connect each site network to another. For example:

HASP terminals can have a maximum of 22 devices.

3780 terminals can have a maximum of four devices.

The number of devices that can access an X.25 network depends on the number of virtual circuits leased from the X.25 network.

Such constraints are mentioned in the main text where each of the the NDL statements is described.

TABLE F-1. MAXIMUM NUMBER OF ELEMENTS

Element	Maximum Number[†]	
	Per Network	Per Host
Applications	32130	510
Couplers	63	8
Hosts	63	1
NPUs	255	255
Terminals with simultaneous access	257985	4095
Trunks per remote NPU to use for loading	4	4
Users	64449	1023

[†]Minus the number of couplers; this maximum is determined by the range of allowed node numbers.

INDEX

OP parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    X.25 protocol
        DEVICE statement 5-7
        TERMDEV statement 5-4
OPGO parameter 3-2
OUTCALL statement 8-5
Output C-4
Output from program 11-1
Owning console C-4

PA parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    X.25 protocol
        DEVICE statement 5-7
        TERMDEV statement 5-4
Packet Assembly/Disassembly (PAD) 1-5, 5-4, C-4
Packet-Switching Network (PSN) 1-5, C-4
PAD (see Packet Assembly/Disassembly)
PAD parameter
    Site-defined protocol
        TERMDEV statement 7-10
        TERMINAL statement 7-9
    X.25 protocol
        TERMDEV statement 5-4
        TERMINAL statement 5-4
PAD subTIP C-4
PAPPL parameter 8-2
Parameterization C-4
Parameters
    AB
        Asynchronous protocol 4-7, 4-10
        Site-defined protocol 7-10, 7-14
    ABL
        Asynchronous protocol 4-7, 4-10, 4-11
        OUTCALL statement 8-6
        Site-defined protocol 7-10, 7-14, 7-15
        Synchronous protocol 6-7, 6-10, 6-11, 6-20,
                           6-22, 6-23, 6-31, 6-34,
                           6-35, 6-41, 6-42, 6-43
        X.25 protocol 5-4, 5-7
    ACCLEV 8-6
    AL
        Asynchronous protocol 4-2, 4-4
        Site-defined protocol 7-1, 7-6
        Synchronous protocol 6-2, 6-4
        X.25 protocol 5-2
    ANAME 8-10
    ARSPEED
        Asynchronous protocol 4-2, 4-4
        Site-defined protocol 7-1, 7-6
    AUTO
        Asynchronous protocol 4-2, 4-4
        Site-defined protocol 7-1, 7-6
        Synchronous protocol 6-2, 6-4
    AUTOCON
        Asynchronous protocol 4-7, 4-10, 4-11
        Site-defined protocol 7-10, 7-14, 7-15
        Synchronous protocol 6-7, 6-10, 6-11, 6-20,
                           6-22, 6-23, 6-31, 6-34, 6-35
                           6-41, 6-42, 6-43
        X.25 protocol 5-4, 5-7, 5-8

BCF
    Site-defined protocol 7-9, 7-10
    Synchronous protocol 6-31, 6-32
BR
    Asynchronous protocol 4-7, 4-10, 4-11
    Site-defined protocol 7-10, 7-14, 7-15
    X.25 protocol 5-4, 5-7, 5-8
BS
    Asynchronous protocol 4-7, 4-10, 4-11
    Site-defined protocol 7-10, 7-14, 7-15
    X.25 protocol 5-4, 5-7, 5-8
B1
    Asynchronous protocol 4-7, 4-10, 4-12
    Site-defined protocol 7-10, 7-14, 7-16
    Synchronous protocol 6-7, 6-10, 6-11, 6-20,
                           6-22, 6-23, 6-41, 6-42, 6-43
    X.25 protocol 5-4, 5-7, 5-8
B2
    Asynchronous protocol 4-7, 4-10, 4-12
    Site-defined protocol 7-10, 7-14, 7-16
    Synchronous protocol 6-7, 6-10, 6-12, 6-20,
                           6-22, 6-24, 6-41, 6-42, 6-44
    X.25 protocol 5-4, 5-7, 5-9
CA
    Site-defined protocol 7-9, 7-10, 7-11
    Synchronous protocol 6-7, 6-8, 6-41
CI
    Asynchronous protocol 4-7, 4-10, 4-13
    Site-defined protocol 7-10, 7-14, 7-16
    X.25 protocol 5-4, 5-7, 5-9
CN
    Asynchronous protocol 4-7, 4-10, 4-13
    Site-defined protocol 7-10, 7-14, 7-16
    Synchronous protocol 6-7, 6-10, 6-12, 6-20,
                           6-22, 6-24, 6-41, 6-42, 6-44
    X.25 protocol 5-4, 5-7, 5-9
CO
    Site-defined protocol 7-9, 7-10, 7-12
    Synchronous protocol 6-20, 6-21, 6-31,
                           6-33
COLLECT
    INCALL statement 8-10
    Site-defined protocol 7-9, 7-10, 7-12
    X.25 protocol 5-4, 5-5
CP
    Asynchronous protocol 4-7, 4-10, 4-13
    Site-defined protocol 7-10, 7-14, 7-16
    X.25 protocol 5-4, 5-7, 5-10
CSET
    Asynchronous protocol 4-7, 4-8
    Site-defined protocol 7-9, 7-10, 7-12
    Synchronous protocol 6-7, 6-9, 6-20, 6-21,
                           6-31, 6-33, 6-41, 6-42
    X.25 protocol 5-4, 5-5, 5-21, 5-22
CT
    Asynchronous protocol 4-7, 4-10, 4-14
    Site-defined protocol 7-10, 7-14, 7-17
    Synchronous protocol 6-7, 6-10, 6-13, 6-20,
                           6-22, 6-25, 6-31, 6-34, 6-35
                           6-41, 6-42, 6-45
    X.25 protocol 5-4, 5-7, 5-10
DBL
    Asynchronous protocol 4-7, 4-10, 4-14
    INCALL statement 8-10
    OUTCALL statement 8-6
    Site-defined protocol 7-10, 7-14, 7-17
    Synchronous protocol 6-7, 6-10, 6-13, 6-20,
                           6-22, 6-25, 6-31, 6-34,
                           6-36, 6-41, 6-42, 6-45
    X.25 protocol 5-4, 5-7, 5-10
DBZ
    Asynchronous protocol 4-7, 4-10, 4-14
    INCALL statement 8-10

TERMINAL statement
    Asynchronous protocol 4-7
    General definition 3-9
    Site-defined protocol 7-9
    Synchronous protocol 6-7, 6-20, 6-31, 6-41
    X.25 protocol 5-4, 5-21
Terminal user C-6
Terminal Verification Facility (TVF) 1-12, 8-3
TIP (see Terminal Interface Program)
TIPTYPE parameter
    Asynchronous protocol
        GROUP statement 4-4
        LINE statement 4-2
    Site-defined protocol
        GROUP statement 7-6
        LINE statement 7-1
    Synchronous protocol
        GROUP statement 6-4
        LINE statement 6-2
    X.25 protocol
        LINE statement 5-2
TITLE statement 2-5
Transaction Facility (TAF) 1-12, 8-3
Transmission block C-6
Transparent mode C-6
Trunk 1-8, C-6
TRUNK statement 3-6
TSPEED parameter
    Asynchronous protocol
        TERMDEV statement 4-7
        TERMINAL statement 4-7
    Site-defined protocol
        TERMDEV statement 7-10
        TERMINAL statement 7-9
TVF (see Terminal Verification Facility)

UBL parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    INCALL statement 8-10
    OUTCALL statement 8-6
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    Synchronous protocol
        DEVICE statement 6-10, 6-22, 6-34, 6-42
        TERMDEV statement 6-7, 6-20, 6-31, 6-41
    X.25 protocol
        DEVICE statement 5-7
        TERMDEV statement 5-4
UBZ parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    INCALL statement 8-10
    OUTCALL statement 8-6
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    Synchronous protocol
        DEVICE statement 6-10, 6-22, 6-34, 6-42
        TERMDEV statement 6-7, 6-20, 6-31, 6-41
    X.25 protocol
        DEVICE statement 5-7
        TERMDEV statement 5-4
UDATA parameter 8-6
UID parameter 8-4
UNAME parameter 8-10

Upline
    Block C-6
    Data 1-9
    Description C-6
User name C-6
USER statement 8-1

VALIDUz 1-2
Value declaration format 2-1
Value-required keywords 2-1
Variant 1-4
VARIANT parameter 3-2
VEIAF (see NOS/VE Interactive Facility)
Virtual circuit 1-6

W parameter
    Site-defined protocol
        TERMDEV statement 7-10
        TERMINAL statement 7-9
    X.25 protocol
        TERMDEV statement 5-4
        TERMINAL statement 5-4
WR parameter 8-10
WS parameter
    INCALL statement 8-10
    OUTCALL statement 8-6

XAUTO parameter
    Asynchronous protocol
        GROUP statement 4-4
        LINE statement 4-2
    Site-defined protocol
        GROUP statement 7-6
        LINE statement 7-1
XBZ parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    Synchronous protocol
        DEVICE statement 6-10, 6-22, 6-34
        TERMDEV statement 6-7, 6-20, 6-31
XLC parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    X.25 protocol
        DEVICE statement 5-7
        TERMDEV statement 5-4
XLTO parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7
    Site-defined protocol
        DEVICE statement 7-14
        TERMDEV statement 7-10
    X.25 protocol
        DEVICE statement 5-7
        TERMDEV statement 5-4
XLX parameter
    Asynchronous protocol
        DEVICE statement 4-10
        TERMDEV statement 4-7

COMMENT SHEET

MANUAL TITLE: Network Access Method Version 1 Network Definition Language
                Reference Manual

PUBLICATION NO.: 60480000

REVISION: S

This form is not intended to be used as an order blank. Control Data Corporation
welcomes your evaluation of this manual. Please indicate any errors, suggested
additions or deletions, or general comments on the back (please include page number
references).

_____ Please reply   _____ No reply necessary

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.
FOLD ON DOTTED LINES AND TAPE

NAME:

COMPANY:

STREET ADDRESS:

CITY/STATE/ZIP:

CUT ALONG LINE

TAPE   TAPE

CD CONTROL DATA